UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/681,398	03/29/2001	Steve C. Wang	200-1381	3286

28804 7590 08/16/2004

CHUPA & ALBERTI, P.C.
31313 NORTHWESTERN HWY
SUITE 205
FARMINGTON HILLS, MI 48334

EXAMINER	
BRODA, SAMUEL	
ART UNIT	**PAPER NUMBER**
2123	

DATE MAILED: 08/16/2004

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary	Application No.	Applicant(s)	
	09/681,398	WANG ET AL.	
	Examiner	Art Unit	
	Samuel Broda	2123	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If the period for reply specified above is less than thirty (30) days, a reply within the statutory minimum of thirty (30) days will be considered timely.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *29 March 2001*.

2a)☐ This action is **FINAL**. 2b)☒ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-19* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-19* is/are rejected.

7)☒ Claim(s) *1-19* is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☒ The drawing(s) filed on *29 March 2001* is/are: a)☒ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☐ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☒ Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08) Paper No(s)/Mail Date *4/2001*.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____ .
5) ☐ Notice of Informal Patent Application (PTO-152)
6) ☒ Other: *Request for Information*.

DETAILED ACTION

1. Claims 1-19 have been examined.

Drawings

2. Applicants' formal drawings have been reviewed and approved by the PTO Draftsperson.

Claim Objections

3. The following is a partial quotation of 37 CFR § 1.75:

> . . .
> (i) Where a claim sets forth a plurality of elements or steps, each element or step of the claim should be separated by a line indentation.

3.1 Claims 1-19 are objected to under 37 CFR § 1.75(i) because each element of each claim is not separated by a line indentation.

Claim Rejections - 35 U.S.C. § 112, Second Paragraph

4. The following is a quotation of the second paragraph of 35 U.S.C. 112:

> The specification shall conclude with one or more claims particularly pointing out and distinctly claiming the subject matter which the applicant regards as his invention.

4.1 Claims 12-19 are rejected under 35 U.S.C. 112, second paragraph, as being indefinite for failing to particularly point out and distinctly claim the subject matter which Applicants regard as the invention.

4.2 Regarding independent claim 12, the claim preamble is directed to a "method for analyzing a design" but the method steps appear directed to mathematical operations for approximating a performance surface. Because none of the method steps appear connected to analyzing a design, the scope of the claim is unclear.

4.3 Claims 13-19 are dependent on claim 12 and rejected using the same analysis.

Claim Rejections - 35 U.S.C. § 101

5. The following is a quotation of 35 U.S.C. 101:

> Whoever invents or discovers any new and useful process, machine, manufacture, or composition of matter, or any new and useful improvement thereof, may obtain a patent therefor, subject to the conditions and requirements of this title.

5.1 Method claims 7-19 are rejected for reciting a process comprising an abstract idea.

5.2 Regarding independent claim 7, this claim is directed to "a method for analyzing a computer generated model," and the steps recited in claim 7 describe the abstract idea of performing a mathematical analysis on a variable. Regarding independent claim 12, this claim is directed to "a method for analyzing a design," and the steps recited in claim 12 describe the abstract idea of performing mathematical operations on a performance surface.

The steps recited in these claims do not: (1) recite data gathering limitations or post-mathematical operations that might independently limit the claims beyond the performance of a

mathematical operation; or (2) limit the use of the output to a practical application providing a useful, concrete, and tangible result.

5.3 Dependent claims 8-11 and 13-19 are rejected using the same analysis.

Claim Rejections - 35 U.S.C. § 102

6. The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

> A person shall be entitled to a patent unless –
> (a) the invention was known or used by others in this country, or patented or described in a printed publication in this or a foreign country, before the date of invention thereof by the applicant for patent.

6.1 Claims 1-11 are rejected under 35 U.S.C. 102(a) as being anticipated by Ye et al, "Algorithmic Construction of Optimal Symmetric Latin Hypercube Design, " Journal of Statistical Planning and Inference, Vol. 90 No. 1, pp. 145-159 (July 2000).

6.2 Regarding claim 7, Ye et al teaches a method for analyzing a computer generated model, including:

receiving the computer generated model [model of diameter of a cyclone, page 22 equation (3);

creating at least one variable [seven variables used to predict the diameter of the cyclone; see page 22]; and

probabilistically analyzing the computer generated model by the use of the at least one variable [range of inputs used in modified Latin Hypercube Designs and mean squared error for each design, Table 10 and corresponding text at pages 22-25].

6.3 Regarding claims 8-9, the method of Ye et al samples a performance surface corresponding to 400 randomly selected sites and a calculated mean squared error to analyze the design, with the determination of the sampled performance space adequately approximated by the lowest mean squared error. See Table 10 and corresponding text.

6.4 Regarding claims 10-11, the method of Ye et al samples a performance surface corresponding to mean squared error to analyze the design, with the determination of the sampled performance space adequately approximated by the lowest mean squared error. See Table 10 and corresponding text.

Additionally, Ye et al teaches establishing a prediction model for both control and noise factors, and then given the distribution of noise variables, estimating the variation of the response for each combination of control variables using the model. See page 25 paragraph 1.

6.5 Regarding system claims 1-4, these claims correspond to method claims 7-9 and are anticipated using the same analysis.

6.6 Regarding system claims 5-6, these claims correspond to method claims 10-11 and are anticipated using the same analysis.

Conclusion

7. The prior art made of record and not relied upon is considered pertinent to Applicants' disclosure. Reference to Ali et al, U. S. Patent 6,405,344 issued 11 June 2002 and filed 14 May 1999, is cited as teaching a method for performing design trade-off based on a total score.

Reference to Ostrowski et al, U. S. Patent 6,377,908 issued 23 April 2002 and filed 14 May 1999, is cited as teaching a method for optimizing transfer function outputs in which inputs are perturbed.

Reference to Bailey et al, "Using Response Surfaces to Improve the Search for Satisfactory Behavior in System Dynamics Models," System Dynamics Review, Vol. 16 No. 2, pp. 75-90 (2000), is cited as teaching an augmented robust concept exploration method.

Reference to Hamada, "Using Statistically Designed Experiments to Improve Reliability and to Achieve Robust Reliability," IEEE Transactions on Reliability, Vol. 44 No. 2, pp. 206-215 (June 1995), is cited as teaching Taguchi experimental designs.

Reference to Boning et al, "DOE/Opt: A System for Design of Experiments, Response Surface Modeling, and Optimization Using Process and Device Simulation," IEEE Transactions on Semiconductor Manufacturing, Vol. 7 No. 2, pp. 233-244 (May 1994), is cited as teaching the DOE/Opt system that integrates design of experiments, response surface model generation, and nonlinear constrained optimization.

8. This Office action has an attached requirement for information under 37 CFR 1.105. A complete reply to this Office action must include a complete reply to the attached requirement

for information. The time period for reply to the attached requirement coincides with the time period for reply to this Office action.

9. Any inquiry concerning this communication or earlier communications from the Examiner should be directed to Samuel Broda, whose telephone number is (703) 305-1026. The Examiner can normally be reached on Mondays through Fridays from 8:00 AM – 4:30 PM.

If attempts to reach the Examiner by telephone are unsuccessful, the Examiner's supervisor, Kevin Teska, can be reached at (703) 305-9704. The fax phone number for the organization where this application or proceeding is assigned is (703) 872-9306.

Any inquiry of a general nature or relating to the status of this application or proceeding should be directed to the group receptionist, whose telephone number is (703) 305-3900.

SAMUEL BRODA, ESQ.
PRIMARY EXAMINER

REQUIREMENT FOR INFORMATION UNDER 37 C.F.R. § 1.105

ATTACHMENT TO PAPER NO. 200408

1. Applicants and the assignee of this application are required under 37 CFR 1.105 to provide the following information that the examiner has determined is reasonably necessary to the examination of this application.

2. In response to this requirement, please provide copies of each publication (published prior to the filing date) which any of the Applicants authored or co-authored and which describe the disclosed subject matter of:

(1) Latin hypercube designs; or

(2) entropy analysis.

3. In responding to those requirements that require copies of documents, where the document is a bound text or a single article over 50 pages, the requirement may be met by providing copies of those pages that provide the particular subject matter indicated in the requirement, or where such subject matter is not indicated, the subject matter found in Applicants' disclosure.

4. The fee and certification requirements of 37 C.F.R. § 1.97 are waived for those documents submitted in reply to this requirement. This waiver extends only to those documents within the scope of this requirement under 37 C.F.R. § 1.105 that are included in the Applicants' first complete communication responding to this requirement. Any supplemental replies

subsequent to the first communication responding to this requirement and any information disclosures beyond the scope of this requirement under 37 C.F.R. § 1.105 are subject to the fee and certification requirements of 37 C.F.R. § 1.97.

5. The applicant is reminded that the reply to this requirement must be made with candor and good faith under 37 CFR 1.56. Where the applicant does not have or cannot readily obtain an item of required information, a statement that the item is unknown or cannot be readily obtained will be accepted as a complete response to the requirement for that item.

6. This requirement is an attachment of the enclosed Office action. A complete response to the enclosed Office action must include a complete response to this requirement. The time period for reply to this requirement coincides with the time period for reply to the enclosed Office action, which is THREE months.

SAMUEL BRODA, ESQ.
PRIMARY EXAMINER

KEVIN J. TESKA
SUPERVISORY
PATENT EXAMINER

PTO/SB/PTO Rev 10/95	U.S. Department of Commerce Patent and Trademark Office	Complete If Known	
LIST OF PRIOR ART CITED BY APPLICANT		Application Number	
		Filing Date	
		First Named Inventor	Jiang et al.
(use as many sheets as necessary)		Group Art Unit	
		Examiner Name	
Sheet 1 of 1		Attorney Docket Number	200-1381

OTHER PRIOR ART—NON PATENT LITERATURE DOCUMENTS			
Examiner Initials*	Cite No.[1]	Include name of the author (in CAPITAL LETTERS), title of the article (when appropriate), title of the item (book, magazine, journal, serial, symposium, catalog, etc.), date, page(s), volume-issue number(s), publisher, country where published, source.	T[2]
AB	1	WILLIAM J. WELCH, ROBERT J. BUCK, JEROME SACKS, HENRY P. WYNN, TOBY J. MITCHELL AND MAX D. MORRIS, Screening, Predicting, and Computer Experiments, Technometrics, February, 1992, pages 15-25, Vol. 34, No. 1, U.S.	
AB	2	JEROME H. FRIEDMAN, Invited Paper, Multivariate Adaptive Regression Splines, Stanford University, 1991, Vol. 19, No. 1, 1-141, The Annals of Statistics, U.S.	

Examiner Signature	Samuel Broda	Date Considered	8/6/04

EXAMINER: Initial if reference considered, whether or not citation is in conformance with MPEP 609. Draw line through citation if not in conformance and not considered. Include copy of this form with next communication to applicant.

1 Unique citation designation number. 2 Applicant is to place a check mark here if English language Translation is attached.

Burden Hour Statement: This form is estimated to take .2 hours to complete. Time will vary depending upon the needs of the individual case. Any comments on the amount of time you are required to complete this form should be sent to the Chief Information Officer, Patent and Trademark Office, Washington, D.C. 20231.
DO NOT SEND FEES OR COMPLETED FORMS TO THIS ADDRESS. Send To: Assistant Commissioner for Patent, Washington, D.C. 20231.

(Information Disclosure Statement – Section 2. FORM 1449A/PTO [6-1] – page 1 of 1)

John G. Chupa (Reg. No. 33,483)
Chupa & Alberti, P.C.
31313 Northwestern Highway
Suite 205
Farmington Hills, MI 48334

Notice of References Cited	Application/Control No. 09/681,398	Applicant(s)/Patent Under Reexamination WANG ET AL.	
	Examiner Samuel Broda	Art Unit 2123	Page 1 of 1

U.S. PATENT DOCUMENTS

*		Document Number Country Code-Number-Kind Code	Date MM-YYYY	Name	Classification
	A	US-6,405,344 B1	06-2002	Ali et al.	716/2
	B	US-6,377,908 B1	04-2002	Ostrowski et al.	703/2
	C	US-			
	D	US-			
	E	US-			
	F	US-			
	G	US-			
	H	US-			
	I	US-			
	J	US-			
	K	US-			
	L	US-			
	M	US-			

FOREIGN PATENT DOCUMENTS

*		Document Number Country Code-Number-Kind Code	Date MM-YYYY	Country	Name	Classification
	N					
	O					
	P					
	Q					
	R					
	S					
	T					

NON-PATENT DOCUMENTS

*		Include as applicable: Author, Title Date, Publisher, Edition or Volume, Pertinent Pages)
	U	Ye et al, "Algorithmic Construction of Optimal Symmetric Latin Hypercube Design, " Journal of Statistical Planning and Inference, Vol. 90 No. 1, pp. 145-159 (July 2000)
	V	Bailey et al, "Using Response Surfaces to Improve the Search for Satisfactory Behavior in System Dynamics Models," System Dynamics Review, Vol. 16 No. 2, pp. 75-90 (2000)
	W	Hamada, "Using Statistically Designed Experiments to Improve Reliability and to Achieve Robust Reliability," IEEE Transactions on Reliability, Vol. 44 No. 2, pp. 206-215 (June 1995)
	X	Boning et al, "DOE/Opt: A System for Design of Experiments, Response Surface Modeling, and Optimization Using Process and Device Simulation," IEEE Transactions on Semiconductor Manufacturing, Vol. 7 No. 2, pp. 233-244 (May 1994)

*A copy of this reference is not being furnished with this Office action. (See MPEP § 707.05(a).)
Dates in MM-YYYY format are publication dates. Classifications may be US or foreign.

Algorithmic construction of optimal symmetric Latin hypercube designs

Kenny Q. Ye	William Li	Agus Sudjianto
Department of Applied Mathematics and Statistics	Department of Operations and Management Science	Reliability Methods Department
SUNY at Stony Brook	University of Minnesota	Ford Motor Company
Stony Brook, NY 10021	Minneapolis, MN 55455	Dearborn, MI 48121

Abstract

We propose symmetric Latin hypercubes for designs of computer exepriment. The goal is to offer a compromise between computing effort and design optimality. The proposed class of designs has some advantages over the regular Latin hypercube design with respect to criteria such as entropy and the minimum intersite distance. An exchange algorithm is proposed for constructing optimal symmetric Latin hypercube designs. This algorithm is compared

with two existing algorithms by Park (1994) and Morris and Mitchell (1995). Some examples, including a real case study in the automotive industry, are used to illustrate the performance of the new designs and the algorithms.

Key Words: Computer experiment; Maximum Entropy Design; Maximin design

1 Introduction

One of our recent projects is concerned with the thermal analysis of multi-layer electrical traces at a major automotive company. As more and more electronic devices are installed in vehicles, the peak temperature of electrical traces becomes a major concern in designing the instrument panels. The temperature of an electrical trace is largely determined by its width, its passing current strength, and its position in a stack of traces. The goal of this project is to provide guidelines for design engineers for width and passing current strength of multi-layer electrical traces. Physical experiments are inevitably very expensive and time consuming since a set of electrical traces has to be assembled in certain configurations for each test and measuring the temperature of each trace is difficult. Therefore, finite element analysis

(FEA) models have been developed to simulate the thermal dynamics of electrical traces.

Using the computer model, the study starts from a simple case, in which there are two layers with three traces on each layer. One primary interest is the interaction between a center trace and an edge trace on two different layers since the heat coming off the center trace spreads out and affects the temperature at the edge. A center trace on layer 1 and an edge trace on layer 2 are selected in the study. The goal is to predict their peak temperatures (y_1 and y_2) based on four predictors: the width of the center trace (x_1), the applied current of the center trace (x_2), the width of the edge trace (x_3), and the applied current of the edge trace (x_4). Given a set of x_i-values, the computer model generates a deterministic peak temperature for each trace.

Though computer experiments are much cheaper and faster than physical experiments, each run is still time consuming and expensive. Thus, only a small number of combinations of the x_i can be tested. In this case, the experiment is to be conducted by another company which specializes in thermal dynamic computer models, and the budget only allows for 25 runs. A feasible approach is to establish a statistical model from the results of the 25 runs and then use it to predict peak temperatures for any given combinations of

the x_i. An optimal Latin hypercube design was chosen for this experiment.

A Latin hypercube design (LHD) is an $n \times l$ matrix in which each column is a random permutation of $\{1, \ldots, n\}$ which can be mapped onto the actual range of the variables. It has good projection properties on any single dimension. Latin hypercube designs have been applied in many computer experiments since they were proposed by McKay *et al.* (1979). In practice, a LHD can be randomly generated, but a randomly selected LHD may have bad properties and act poorly in estimation and prediction. Another approach is to use optimal designs according to some criteria such as entropy (Shewry and Wynn 1987). Integrated Mean Squared Error (IMSE) (Sacks et al. 1989), and minimum intersite distance (Johnson *et al.* 1990). These designs have been shown to be efficient for certain models. However, the computational cost of obtaining these designs is high. In an attempt to offer a compromise between good projective properties of LHDs and a criterion, Park (1994) and Morris and Mitchell (1995) proposed optimal Latin hypercube designs. For an excellent review of design and analysis of computer experiments, see Koehler and Owen (1996).

One of the criteria considered in this paper is the entropy criterion, first proposed by Shewry and Wynn (1987) and then adopted by Currin

et al. (1991). The response of a computer model is modeled by $Y(\mathbf{x}) = \sum_{j=1}^{k} \beta_j f_j(\mathbf{x}) + Z(\mathbf{x})$, in which $Z(\mathbf{x})$ is a Gaussian process with mean zero and covariance

$$R(\mathbf{s}, \mathbf{t}) = \sigma^2 \exp\left\{-\theta \sum_{j=1}^{l} |\mathbf{s}_j - \mathbf{t}_j|^q\right\}, \quad 0 < q \leq 2 \tag{1}$$

between two l-dimensional inputs $\mathbf{s}$ and $\mathbf{t}$. The entropy criterion is equivalent to the minimization of $-\log|R|$, where R is the covariance matrix of the design. The parameters θ and q determine the properties of $Z(x)$. Throughout this paper, we set $q = 2$ so that the correlation between two sites is a function of their L_2 distance.

The construction of an optimal LHD can still be time consuming. For example, to generate a maximum entropy 25×4 LHD using a columnwise-pairwise (CP) algorithm (discussed in Section 3), the whole procedure takes 3.3 hours on a Sun SPARC 20 workstation, which appears to be quite long as the size of the design is moderate. The search for a larger design would take even longer, and may be computationally prohibitive. This situation motivated us to look for alternatives that require less computing time. The easiest method is to generate a large number of random LHDs and then choose the best one according to the criterion. For example, the generation of 1000 random LHDs takes only 14.7 seconds on the same machine. However,

the best design obtained from these random designs is usually significantly inferior to that produced by the algorithmic search. In our example, the entropy value at $\theta = 2$ of the former is 25.26, compared with the latter's 20.48. To reduce the searching time and still generate competitive designs, our approach is to restrict the search within a subset of the general LHD. If this subset of designs has some desirable properties with respect to a criterion, then selecting a design from this group of designs may be more efficient.

In Section 2, we introduce a new class of LHD, the *symmetric Latin hypercube design*, whose geometric property enables us to find optimal LHDs more efficiently. Section 3 considers a simple exchange algorithm for constructing optimal symmetric LHDs. Its performance is compared with the existing algorithmic approaches of Park (1994), and Morris and Mitchell (1995). Section 4 demonstrates the performance of the new design with an example. A summary is given in Section 5.

2 Symmetric Latin hypercubes

Our goal is to find a special type of LHD that has some good "built-in" properties with respect to the optimality of a design. In our definition, a

LHD is called a *Symmetric Latin hypercube design* (SLHD) if it has the following property: in an $n \times l$ LHD with levels from 1 to n, if $(a_1, a_2, \cdots, a_l)$ is one of the rows, then the vector $(n+1-a_1, n+1-a_2, \cdots, n+1-a_l)$ must be another row in the design matrix. In other words, if $\mathbf{t}_i$ is a design point in a SLHD, then there exists another point $\mathbf{t}_j$ in the design that is the reflection of $\mathbf{t}_i$ through the center. An example of a 10×5 SLHD is given in Table 1, in which the i^{th} row is the symmetric point of the $(n+1-i)^{th}$ row.

1	2	3	4	5
1	6	6	5	9
2	2	3	2	4
3	1	9	7	5
4	3	4	10	3
5	7	1	8	10
6	4	10	3	1
7	8	7	1	8
8	10	2	4	6
9	9	8	9	7
10	5	5	6	2

Table 1: A 10×5 symmetric Latin hypercube design

The symmetry of a SLHD provides some orthogonal properties. That is, the estimation of the linear effect of each variable in a SLHD is uncorrelated with all quadratic effects and bi-linear interactions. The proof of such properties can be found in Ye (1998), in which Orthogonal Latin Hypercube

Designs (OLHD) are constructed and proposed. OLHDs have the same symmetric properties but also process additional orthogonality which insures the zero correlation among estimation of linear effects. Therefore, one can view the SLHD as a generalization of the OLHD. However, the number of runs in an OLHD has to be a power of two, which increases dramatically as the number of factors increases. In the cases that an appropriate OLHD can not be found under the constraint of run size, one can consider using SLHDs which have the flexibility of the run size, yet retain some of the orthogonality of an OLHD.

Are SLHDs better than regular LHDs with respect to design criteria? Many optimal LHDs reported by Park (1994) and Morris and Mitchell (1995) have some symmetric properties. In particular, Morris and Mitchell (1995) noticed that a large number of the optimal LHDs they obtained are SLHDs and referred them as "foldover designs". This is the first time the SLHD is mentioned in the literature. They called for a thorough investigation on this phenomenon. Intuitively, the optimal designs are considered to have good space filling properties, and a good space filling design probably has some degree of symmetry. To verify this, we undertook a simulation study to compare random SLHDs to random LHDs with respect to both entropy and

minimum intersite distance criteria. Table 2 compares the best design among the 1000 random SLHDs of size 25 with that of the 1000 random regular LHDs. The former has a smaller entropy criterion value of 23.60 at $\theta = 2$, compared with the latter's 25.26. In the table, we also list the minimum distances of three designs, a criterion first proposed by Johnson *et al.* (1990) and then used by Morris and Mitchell (1995) in constructing optimal LHDs. For a design S, the minimum distance $d^*(S) = \min_{\mathbf{s},\mathbf{t}\in S} d(\mathbf{s},\mathbf{t})$, where $\mathbf{s}$ and $\mathbf{t}$ are two design points (*i.e.*, two rows in the design). Both the L_1 (rectangular) distance $L_1(\mathbf{s},\mathbf{t}) = \sum_{j=1}^{l} |\mathbf{s}_j - \mathbf{t}_j|$ and L_2 (Euclidean) distance $L_2(\mathbf{s},\mathbf{t}) = [\sum_{j=1}^{l}(\mathbf{s}_j - \mathbf{t}_j)^2]^{\frac{1}{2}}$ of three designs are listed in Table 2. A design S_1 is said to be better than design S_2 if $d^*(S_1) > d^*(S_2)$. The number of pairs separated by this distance, denoted J, is shown in parentheses in the table. If two designs have the same d^* values, then the design with smaller J value is better. Throughout this paper, entropy and distances are computed after LHDs are scaled into $[0,1]^l$. The levels of scaled LHDs are $\{0, \frac{1}{n-1}, \frac{2}{n-1}, \cdots, 1\}$, which are the same used by Morris and Mitchell (1995). Note that Park (1994) used a different scale, $\{\frac{1}{n}, \frac{2}{n}, \cdots, 1\}$.

Tables 3 and 4 provide more comparisons between regular LHDs and SLHDs. We study Latin hypercubes with six different sizes, 25×4, 20×6,

design	entropy	min. L_1 distance	min. L_2 distance
best random LHD	25.26	0.50 (1)	0.29 (1)
best random SLHD	23.60	0.63 (2)	0.35 (2)
optimal LHD	20.48	0.75(1)	0.50(1)

Table 2: Comparison of three 25×4 LHDs by (i) entropy criterion($\theta = 2$); (ii) smallest L_1 distance and the number of pairs separated by that distance (in parentheses); and (iii) smallest L_2 distance and the number of pairs separated by that distance. The "best" is in terms of the entropy criterion.

	LHD		SLHD	
Size	Mean	Max	Mean	Max
25×4	0.3478	0.5417	0.3944	0.625
20×6	0.8205	1.211	0.8968	1.316
50×10	1.316	1.735	1.407	1.816
200×20	2.9457	3.4925	3.0884	3.5678
500×30	4.8737	5.3567	5.0240	5.4188
1000×50	9.4014	9.9339	9.6016	10.2002

Table 3: Minimum L_1 distances of random SLHDs and random LHDs

50×10 , 200×20, 500×30, 1000×50. The sample sizes are 1000 for the first three designs and 100 for the last two designs. SLHDs are consistently superior to the corresponding LHDs with respect to both L_1 and L_2 distances. We also compare the means of minimum distance of LHDs and SLHDs using t-tests. The p-values are all smaller than 0.0001. Therefore, SLHDs are statistically significantly better than LHDs with respect to the minimum distance criteria.

	LHD		SLHD	
Size	Mean	Max	Mean	Max
25×4	0.1943	0.3200	0.2230	0.3727
20×6	0.3875	0.5642	0.4270	0.6316
50×10	0.5163	0.7061	0.5492	0.7210
200×20	0.8492	1.0568	0.8879	1.0281
500×30	1.1383	1.2812	1.1812	1.2960
1000×50	1.7327	1.8328	1.7658	1.8689

Table 4: Minimum L_2 distances of random SLHDs and random LHDs

These simulation studies have shown the advantages of "picking the winner" from SLHDs instead of regular LHDs. However, the best SLHD obtained by the "picking the winner" approach is usually inferior to the corresponding optimal design obtained by a searching algorithm, as shown in Table 2. In the next section, a simple exchange algorithm is presented to search optimal SLHDs.

3 An algorithm and examples

In this section, we review the two existing algorithms proposed by Park (1994) and Morris and Mitchell (1995), respectively, for constructing optimal LHDs. Then a columnwise-pairwise exchange algorithm (CP) is introduced in the context of the construction of optimal SLHDs. The similarities and differences between the CP and the other two algorithms are discussed.

Through examples, we compare

(1) the performance of the CP and other two algorithms;

(2) the optimal SLHDs and the optimal regular LHDs with respect to the design criteria and the searching time.

3.1 Existing algorithms

To construct optimal LHDs, Park (1994) presented an approach based on the the exchanges of several pairs of the elements in two rows. His algorithm first selects some active pairs which minimize the objective criterion value by excluding that pair from the design. Then for each chosen pair of two rows i_1 and i_2, the algorithm considers all the possible exchanges $x_{i_1 j_1} \leftrightarrow x_{i_2 j_1}, \ldots, x_{i_1 j_k} \leftrightarrow x_{i_2 j_k}$ for $k \leq l$ and finds the best exchange among them.

Morris and Mitchell(1995) adopted a simulated annealing algorithm to search for optimal LHDs. They also defined a maximin distance criterion. For a given design, define a distance list $\{d_1, d_2, \ldots, d_m\}$, $d_1 < d_2 < \ldots < d_m$, in which the d_i's are the distinct values of intersite distances. Let J_i be the number of pairs of sites in the design separated by d_i. Then a design is a maximin distance design if and only if

(1a) d_1 is maximized, and among the designs for which this is true,

(1b) J_1 is minimized, and among the designs for which this is true,

(2a) d_2 is maximized, and among the designs for which this is true,

(2b) J_2 is minimized,

and so forth. Morris and Mitchell (1995) also pointed out that although this extended definition is intuitively appealing, it would be better to use a scalar-valued criterion as the driving criterion. For this purpose, they proposed a family of functions

$$\phi_p = \left[\sum_{j=1}^{m} J_j d_j^{-p}\right]^{1/p}, \tag{2}$$

where p is a positive integer. Normally, different p values are tried to obtain a maximin distance LHD.

In Morris and Mitchell's algorithm, a search begins with a randomly chosen LHD, and proceeds through examination of a sequence of designs, each generated as a perturbation of the preceding one. A perturbation D_{try} of a design D is generated by interchanging two randomly chosen elements within a randomly chosen column in D. The perturbation D_{try} replaces D if it leads to an improvement. Otherwise, it will replace D with probability $\pi = \exp\{-[\phi(D_{\text{try}}) - \phi(D)]/t\}$, where t is a preset parameter known as the "temperature".

3.2 Our algorithm

Li and Wu (1997) considered a class of columnwise-pairwise algorithms in the context of the construction of optimal supersaturated designs. A columnwise algorithm makes exchanges on the columns in a design and can be particularly useful for designs that have structure requirements on the columns. Note that each column in an n-run LHD is a permutation of $\{1, \ldots, n\}$. At each step, another permutation of $\{1, \ldots, n\}$ is chosen to replace a column so that the Latin hypercube structure is retained. Therefore, we adopt the columnwise-pairwise idea in searching for optimal LHDs. However, one important change has to be made to accommodate the special structures of the SLHD. For a SLHD *two simultaneous* pair exchanges are made in each column to retain the symmetry. For example, suppose a column in a 6-row SLHD is $(1, 2, 3, 4, 5, 6)'$. If element 1 is exchanged with i, element 6 must be exchanged with $n + 1 - i$ (i.e. $7 - i$) to keep the design symmetric. The only exception is when element i is exchanged with element $n + 1 - i$, which does not require a second exchange. The exchange procedure for a SLHD with an odd number of rows is slightly different. The center point of the design does not participate in the exchange. For example, if a column in a 7-row SLHD is $(1, 2, 3, 4, 5, 6, 7)'$, then element 4 may not to be exchanged with any other

element.

The algorithm for searching optimal SLHD is summarized as follows:

1. Start with a random SLHD.

2. Each iteration has l steps. At the ith step, the best two simultaneous exchanges within column i are found. The design matrix is updated accordingly.

3. If the resulting design is better with respect to the criterion, repeat Step 2. Otherwise, it is considered to be an "optimal design", and the search is terminated.

The resulting optimal designs depend largely on the starting designs used in the algorithm. Hence, one should repeat the algorithm with several different random starting designs. The best design among the generated optimal designs is chosen to be the final design.

3.3 Examples

Example 1 *CP vs. Simulated Annealing*

The simulated annealing algorithm proposed by Morris and Mitchell (1995) aims at constructing optimal regular LHDs. We modify their algorithm to

search for SLHDs. Similarly, the CP algorithm discussed in the previous section is modified to construct optimal regular LHDs. Both algorithms are columnwise-pairwise procedures. The simulated annealing algorithm operates on a (randomly chosen) column and then considers a (randomly chosen) pair in each column. Our proposed CP algorithm resembles the former with a very low starting temperature (so that switches to inferior designs are never made). An important difference is that the simulated annealing algorithm perturbs the design in a random manner, and our CP algorithm perturbs the design in a deterministic manner.

To compare their performances, we use both algorithms to construct optimal regular LHDs and optimal SLHDs. Two examples are considered. Table 5 lists the 12×2 maximin distance LHDs and SLHDs generated by both algorithms. The simulated annealing algorithm uses 10 starting designs and the CP uses 100 starting designs. The driving criterion is ϕ_p with $p = 50$. To compare the efficiency of the algorithms, the number of searched LHSs is also recorded. Both algorithms obtain equally good optimal designs. But the CP algorithm searches far fewer LHDs than the simulated annealing algorithm does. When both algorithms are used to construct 25×4 designs, the simulated annealing algorithm produces better optimal designs than the CP.

design	algorithm	min. dist.	# of searched LHDs
LHD	Simulated Annealing	.4545 (16)	269520
LHD	CP	.4545 (16)	44220
SLHD	Simulated Annealing	.4545 (16)	240416
SLHD	CP	.4545 (16)	14652

Table 5: Comparison of optimal 12×2 LHDs and SLHDs using two algorithms, CP (100 starting designs) and simulated annealing (10 starting designs). The search criterion is ϕ_p with $p = 50$ and L_2 distance. Note: using the simulated annealing search in LHD, only two out of 10 starting designs result in 0.4545 (16), compared to all 10 in the SLHD case.

design	algorithm	min. L_1 dist.	# of searched LHDs
LHD	Simulated Annealing	.9177 (19)	1537663
LHD	CP	.8750 (6)	2241900
SLHD	Simulated Annealing	.9583 (36)	1426985
SLHD	CP	.9177 (6)	546480

Table 6: Comparison of optimal 25×4 LHDs and SLHDs using two algorithms, CP (100 starting designs) and simulated annealing (10 starting designs). The search criterion is ϕ_p with $p = 50$ and L_1 distance.

Therefore, we may conclude that the systematic search algorithm is better for small designs and the simulated annealing algorithm is better for larger designs.

Example 2 *CP vs. Park*

Park's algorithm (1994) cannot be easily modified to accommodate the property of symmetry. Thus, its comparison with the CP is done through

construction of the optimal 9×2 regular LHDs, which is discussed in detail by Park(1994) to illustrate his exchange algorithms. The CP algorithm is also used to construct 9×2 SLHDs. Table 7 compares the optimal designs generated by two algorithms with respect to the entropy criterion ($\theta = 1, 5, 25$), along with the total number of searched LHDs for each algorithm. Three interesting observations are apparent in this example:

1. The CP seems to consistently produce better LHDs than Park's algorithm with respect to entropy. The former also reaches the final design slightly earlier since it searches fewer LHDs. In fact, exhaustive searches reveal that the CP produces the global optimum for each value of $\theta = 1, 5, 25$. Our study of constructing LHDs of different sizes shows the same patterns.

2. Comparisons between optimal LHDs and the corresponding SLHDs show that the former are slightly better than the latter but take approximately 4 times as much time to search. However, for such a small design, it takes so little time (6 to 7 seconds on a Sun Sparc 20 workstation) for an exhaustive search in the whole LHD class. Therefore, there is no need to restrict the search within SLHD class.

3. At $\theta = 25$, the global optimal LHD is symmetric. Moreover, it is also an orthogonal Latin hypercube as constructed algebraically by Ye (1998).

θ	design	algorithm	optimal	average	# of searched LHDs
1	SLHD	CP	20.38	22.16	5488
	LHD	CP	19.16	19.29	20592
	LHD	Park	20.01	20.79	24132
5	SLHD	CP	3.09	4.35	4768
	LHD	CP	2.95	3.06	20052
	LHD	Park	3.42	3.79	23970
25	SLHD	CP	0.49×10^{-2}	0.83×10^{-2}	4784
	LHD	CP	0.49×10^{-2}	1.11×10^{-2}	19080
	LHD	Park	1.03×10^{-2}	4.67×10^{-2}	23580

Table 7: Comparison of three algorithms for generating 9×2 optimal LHDs with 100 random starting designs

Example 3 *LHD vs. SLHD*

We now revisit the case study at the beginning of this paper: the construction of a 25×4 LHD for the thermal analysis of electrical traces. A primary motivation of using the SLHD is to reduce the searching time. Since the number of possible exchanges of each column in a SLHD is much less than that for a regular LHD, it is expected that the exchange algorithm for the SLHD will use much less CPU time. This is confirmed when the algorithm is applied to the construction of the optimal 25×4 SLHD. Without the restriction to SLHD, it takes 13.3 hours and 10.6 hours for the algorithm to terminate using the entropy and ϕ_p criteria, respectively. Using the same number of starting designs (100), the optimal SLHDs are found only after 1.6

hours and 1.3 hours. The results are summarized in Table 8. Theoretically, the global optimal SLHD cannot be better than the global optimal LHD since the SLHD is a subset of the LHD. It is seen in our Example 2 that the obtained optimal LHDs are globally optimal verified by exhaustive search, but they do not always have the symmetrical structure. Morris and Mitchell (1995) use an exhaustive search to find maximin distance LHDs for many small designs. Not all those global optimal designs are symmetric. In practice, a globally optimal design is rarely obtained when the exhaustive search is not feasible. In our case, with much less searching time, the optimal SLHDs found are actually better than the two optimal LHDs obtained previously with respect to both entropy and minimum distance criteria. The maximum entropy SLHD has the criterion value of 18.53 compared with 20.48 for the previously obtained optimal LHD. The former also has the better (i.e. larger) minimum distance of 0.83, with six pairs separated by this distance. Using ϕ_p as the driving criterion, a SLHD was obtained with four pairs separated by the minimum intersite distance of 0.92, which is considerably better than the optimal maximin distance LHD previously found (six pairs separated by 0.83).

Now we revisit Table 6 and focus on the difference between SLHD and

design	criterion	entropy	min. L_1 dist.	CPU time(hrs)
LHD	entropy	20.48	.75 (1)	13.34
LHD	ϕ_p	23.52	.83 (6)	10.63
SLHD	entropy	18.53	.83 (6)	1.6
SLHD	ϕ_p	19.48	.92 (4)	1.28

Table 8: Comparison of optimal 25×4 LHDs vs. SLHDs with respect to entropy and ϕ_p. The entropy is calculated for $\theta = 2$.

regular LHD. For the simulated annealing algorithm, restricting the search within the SLHD did not save much searching time, but the obtained design is significantly better with respect to the minimum distance criterion. For the CP algorithm, there is a dramatic reduction in searching time after we restrict the search within the SLHDs, yet the obtained design is much better. It is also interesting to observe that in far less time the CP found a better design within the SLHDs than the simulated annealing algorithm found within general LHDs.

4 A robust design simulation example

One of the goals in computer experiments is prediction. The Kriging method was developed in geostatistics and brought into computer experiment by Sacks *et al.* (1989) and Currin *et al.* (1991) to predict untested sites in the

experimental regions. It models the response as a Gaussian process. Given a correlation function of the process, the best linear unbiased predictor of y at site x is given by

$$\hat{y}(x) = \hat{\mu} + \mathbf{r}^T R^{-1}(\mathbf{y} - \hat{\mu}\mathbf{1}),$$

where $\mathbf{r}$ is the vector of correlations between x and the design sites x_i, $\mathbf{R}$ is the correlation matrix among design sites, and $\mathbf{y}$ is the vector of the observed responses. In this section, an example is used to illustrate the advantages of using optimal SLHDs for Kriging methods.

The example presented here is taken from Mori (1985), which was originally presented as a robust design case study. It was later used by Li and Wu (1999) to illustrate an integrated approach to parameter design and tolerance design. The original study is concerned with the design of cyclones, which are used to separate solid mass and gaseous mass in chemical engineering. There are seven variables whose original values are given by $(x_1, x_2, x_3, x_4, x_5, x_6, x_7) = (0.1, 0.3, 0.1, 0.11.5, 16.0, 0.75)$. The relation between the response, the diameter of a cyclone (y), and these seven variables is

$$y = 174.42(\frac{x_1}{x_5})(\frac{x_3}{x_2 - x_1})^{0.85} \times \sqrt{\frac{1 - 2.62\{1 - 0.36(\frac{x_1}{x_2})^{-0.56}\}^{3/2}(\frac{x_1}{x_2})^{1.16}}{x_6 x_7}}. \quad (3)$$

The range of the input variables in this example is taken to be $1 \pm 10\%$ of

the original value (see Table 9).

Unscaled Input	Lower limit	Upper limit
x_1	0.09	0.11
x_2	0.27	0.33
x_3	0.09	0.11
x_4	0.09	0.11
x_5	1.35	1.65
x_6	14.4	17.6
x_7	0.675	0.825

Table 9: Range of the inputs in equation (3)

Experiments with 16 runs are performed using (1) 100 random LHDs; (2) 100 random SLHDs; (3) maximum entropy SLHDs generated by the CP with $\theta = 0.05, 0.5, 1$; and (4) a maximin L_2 distance SLHD generated by the CP with ϕ_{50}. In each experiment, using the Kriging method with the correlation function given in equation (1) with $\theta = 0.05, 0.1, 0.5$, we predict $Y(x)$ at the same 400 randomly selected sites. The mean squared error (MSE) of predictions at these 400 sites was calculated for each experiment. The results are summarized in Table 10. First, it can be seen that the MSE is sensitive to the θ used in the Kriging model but is insensitive to the optimal design criterion. Second, all of the optimal designs are better than the random designs. Third, in this case, SLHDs do not always outperform LHDs.

In practice, the choice of correlation function in the Kriging model is

	Correlation parameter in kriging model		
Design	$\theta = 0.05$	$\theta = 0.1$	$\theta = 0.5$
random SLHD (mean of 100)	0.022	0.027	0.058
random LHD (mean of 100)	0.024	0.026	0.052
Max. Entropy($\theta = 0.05$) SLHD	0.016	0.017	0.025
Max. Entropy($\theta = 0.5$) SLHD	0.017	0.018	0.026
Max. Entropy($\theta = 1$) SLHD	0.019	0.020	0.027
Maximin Distance SLHD	0.020	0.020	0.028

Table 10: Square root of MSE for Maximum Entropy SLHD, Maximin distance SLHD, random LHD and SLHD over 400 randomly selected reference sites

complicated and crucial to prediction accuracy. Sacks *et al.* (1989) suggested using maximum likelihood estimate of θ. However, the modeling process should not be limited to Kriging. One advantage of using Latin hypercube designs is that they can facilitate almost any kind of model, parametric and non-parametric. Authors of this paper have used MARS (multivariate adaptive spline regression), GAM (generalized additive models) and second order polynomials to analyze computer experiments.

The cyclone study was originally a case study in robust design. We choose this study to demonstrate the link between computer experiments and robust designs. Robust design studies can also be carried out using computer models as presented by Welch *et al.* (1990). Orthogonal Latin Hypercube design and Symmetric Latin Hypercube design can be used in a robust design

study as well. One can follow the response model approach of robust designs as proposed in Welch *et al.* (1990) and Shoemaker, Tsui and Wu (1991). First, establish a prediction model for both control and noise factors. Then, given the distribution of noise variables, estimate the variation of Y for each combination of control variables using the model obtained at the first stage. If a computer experiment is not expensive, one can skip the first step and estimate the variation caused by the noise variable directly using the computer model. In that case, Latin hypercubes can serve as a sampling mechanism to obtain samples from noise variable, as it was first proposed by McKay *et al.* (1979).

5 Summary and Discussion

This paper proposes a class of symmetric Latin hypercube designs (SLHDs), referred previously by Morris and Mitchell(1995) as "foldover designs", and a new columnwise-pairwise (CP) algorithm for searching optimal design within the SLHD class as well as within the regular LHD.

We summarize the properties of SLHDs as follows.

1. They are a good subset of LHDs with respect to both entropy and

maximin distance criteria (see Tables 2-4).

2. As a generalization of Orthogonal Latin hypercube designs, SLHDs retain some orthogonality. The estimation of quadratic effects and bi-linear interactions is uncorrelated with the estimation of linear effects.

3. The searching time of the CP algorithm is greatly reduced by restricting the search within the SLHDs (see Tables 5-8). The restriction does not significantly reduce the searching time of the simulated annealing algorithm, but it often leads to better designs (See Table 8).

4. The global optimal LHD is not always a SLHD. Morris and Mitchell (1995) did an exhaustive search to find the optimal LHDs of small sizes. Not all of the true optimal designs they found are symmetric.

Despite the fact that the true optimal LHDs do not necessarily fall into the symmetric class. We recommend using the SLHD in computer experiments for two reasons. First, users will benefit from the orthogonal properties of SLHD as summarized above when they try to fit the data with a polynomial model. A non-symmetric LHD does not have such orthogonality. Second, as shown in Tables 6 and 8, by restricting the search within SLHDs, one could obtain approximately optimal designs in a more efficient manner for

moderate to large-size designs. In fact, in these cases, an exhaustive search is usually prohibitive and one should be less concerned about whether a search method has the potential to reach the global optimum and more about how it can obtain a good design with reasonable computing effort. Especially for computer experiments, extra computing power could be spent on additional runs rather than obtaining a slightly better design.

The performances of the three algorithms for searching optimal LHDs are summarized as follows:

1. The CP algorithm consistently outperforms the algorithm of Park (1994).

2. For smaller designs, the CP algorithm is more efficient than the simulated annealing algorithm of Morris and Mitchell (1995). However, the latter can generate better large designs.

One of the referees suggested that we briefly comment on the performance of optimal LHDs compare to other types of designs proposed for computer experiments in recent literature, such as orthogonal arrays, OA based Latin hypercubes and quasi-Monte Carlo lattices. The comparison of different kind of designs is one of the most important problems and deserves a thorough investigation that is beyond the scope of this paper. However, we would

be glad to share some of our opinions. Unlike traditional designs for which the models are in known forms, the computer experimenter often has little idea which model in his/her statistical toolbox will best describe his/her complex computer model before an experiment is done and several kind of models are tried. Most of the proposed designs for computer experiments allow users to try many different models, linear or nonlinear, parametric or non-parametric. Among those, orthogonal arrays may not be appropriate for computer experiments since they do not take the advantage of flexibility of computer experiment in terms of changing levels. Their projections to low dimensions are only a few points so that they are not good for non-parametric regression methods. However, they are good for fitting low-order polynomial models.

An optimal SLHD actually takes three criteria into consideration: the discrepancy of one-dimension projection optimized by the Latin hypercube structure, desired orthogonality inherited from the symmetric structure, and a third criterion (entropy or minimum distance) optimized through an algorithmic search. Therefore, we expect that optimal SLHD should perform very well with many modeling methods. Quasi-Monte Carlo lattice designs (Fang and Wang, 1994) are generated by some sequence which are asymptotically

optimal in discrepancy measure. Since it spreads the design point evenly in the design space, it should have robust performance with different modeling methods. In particular, a *glp (good lattice point) set* is a Latin Hypercube. Bates *et. al* (1996) compared Latin hypercubes designs with lattice designs and found the quasi-Monte Carlo lattice design performed surprisingly well. Tang(1993) and Owen(1993) proposed a special type of Latin Hypercubes which are constructed based on orthogonal arrays. Such Latin hypercubes spread points evenly on t-dimensional projections. The actual dimension t depends on the strength of the original orthogonal array. However, this approach only provides LHD at the sizes of which orthogonal arrays exist. In Table 11, we compare three LHDs. The first one is the fourth optimal SLHD listed in Table 6. The second one is a *glp* set of generating vector (25;11,29,6,13). The third one is a LHD constructed based on $OA(25; 5^{4-2})$ using the procedure described in Tang (1993). It can be seen that in terms of entropy and minimum intersite distance, the optimal SLH is better than the *glp* and OA-based LH. The *glp*, however, is surprisingly good given the fact that it is easy to generate. Therefore, it could be a good choice if quick solutions to design problems in computer experiments are needed. On the other hand, the OA-based LH is far inferior to the other two designs. Since

	Optimal SLH	glp	OA-based LH
Minimum L_1	0.92(6)	0.75(24)	0.54(13)
Minimum L_2	0.46(2)	0.41(12)	0.29(13)
Entropy $\theta = 0.05$	33.03	39.09	50.17
Entropy $\theta = 1$	21.68	26.99	37.00
Entropy $\theta = 2$	19.82	24.97	34.77
Entropy $\theta = 5$	15.96	20.72	30.02
Entropy $\theta = 10$	13.50	17.96	26.89

Table 11: Comparison of three types of LHDs.

a class of LHD can be constructed based on an orthogonal array, a similar algorithmic approach should be developed to find a better design within the class.

We would like to see more research effort to compare those designs with respect to different performance measure. We think that a good design will not necessarily score the highest for any particular criterion but will be reasonably high for all the criteria.

Acknowledgments

We thank Professors J. S. Park and M. Morris for providing the computer codes of their algorithms. We also thank a referee for pointing out our mistake in calculating the entropy criterion using Park's algorithm. We would also like to thank the associate editor and another referee for encouraging us to include comparison of different types of designs in our paper. We are

also indebted to Professor Morris for his valuable comments on our related research work.

References

Bates, R.A., Buck, R.J., Riccomagno, E., Wynn, H.P. (1996). Experimental Design and Observation for Large Systems, *Journal of Royal Statistical Society*, Series B. **58**, 77-94

Currin, C., Mitchell, T., Morris, D. and Ylvisaker, D. (1991). Bayesian prediction of deterministic functions, with applications to the design and analysis of computer experiments. *Journal of the American Statistical Association* **86**, 953-963.

Fang, K.-T., Wang, Y. (1994) *Number-theoretic Methods in Statistics*, Chapman & Hall, London.

Johnson, M. and Moore, L. and Ylvisaker, D. (1990). Minimax and maximin distance designs. *Journal of Statistical Planning and Inference* **26**, 131-148.

Koehler J. and Owen, A. (1996). Computer experiments. In S. Ghosh and C.R. Rao, editor, *Handbook of Statistics, 13: Design and Analysis of Experiments*, 261-308, 1996, North-Holland.

Li, W. and Wu, C. F. J., (1997). Columnwise-pairwise algorithms with applications to the construction of supersaturated designs. *Technometrics* **39**, 171-179.

Li, W. and Wu, C. F. J., (1999) An intergrated method of parameter design and tolerance design. *Quality Engineering* **11**, 417-425

McKay, M., Beckman, R. and Conover, W. (1979). A comparison of three methods for selecting values of input variables in the analysis of output from a computer code. *Technometrics* **21**, 239-246.

Mori, T. (1985), *Case studies in experimental design*, (in Japanese).

Morris, M. and Mitchell, T. (1995). Exploratory design for computer experiments. *Journal of Statistical Planing and Inference* **43**, 381-402.

Owen, A.(1992), Orthogonal Arrays for Computer Experiments, Integration, and Visualization. *Statistica Sinica*. **2**. 439-452.

Park, J.-S. (1994). Optimal Latin-hypercube designs for computer experiments. *Journal of Statistical Planning Inference* **39**, 95-111.

Sachs, J., Schiller, S. B. and Welch, W. J. (1989). Designs for computer experiments. *Technometrics* **34**, 15-25.

Shewry, M. and Wynn, H. (1987). Maximum entropy design. *Journal of Applied Statistics* **14**, 2, 165-170.

Shoemaker, A. C., Tsui, L. L., and Wu, C. F. J. (1991). Economical experimentation methods for robust design. *Technometrics*, **33**, 415-427.

Tang, B (1993). Orthogonal Array-Based Latin Hypercubes. *Journal of American Statistical Association*, **88**, 1392-1397.

Welch, W. J., Yu, T. K., Kang, S. M. and Sacks, J. (1990). Computer experiments for quality control by parameter design. *Journal of Quality Technology*, **22**, 15-22.

Ye, K. Q.(1998). Column Orthogonal Latin hypercubes and their application in computer experiments. *Journal of American Statistical Association*, **93** 1430-1439.

Algorithmic construction of optimal symmetric Latin hypercube designs (Make Corrections)

Kenny Q. Ye William Li Agus Sudjianto
Department of Applied Department of...

Abstract: We propose symmetric Latin hypercubes for designs of computer exepriment. The goal is to offer a compromise between computing effort and design optimality. The proposed class of designs has some advantages over the regular Latin hypercube design with respect to criteria such as entropy and the minimum intersite distance. An exchange algorithm is proposed for constructing optimal symmetric Latin hypercube designs. This algorithm is compared 1 with two existing algorithms by Park (1994) and... (Update)


```
@misc{ william-algorithmic,
  author = "Kenny Ye William",
  title = "Algorithmic construction of optimal symmetric Latin hyperc
  url = "citeseer.ist.psu.edu/312990.html" }
```

al. - 1991
5 Columnwise-pairwise algorithms with applications to the cons.. (context) - Li, Wu - 1997
3 Optimal Latin-hypercube designs for computer experiments (context) - Park - 1994
2 Exploratory design for computer experiments (context) - Morris, Mitchell - 1995
2 Maximum entropy design (context) - Shewry, Wynn - 1987
2 Case studies in experimental design (context) - Mori - 1985
1 Column Orthogonal Latin hypercubes and their application in .. (context) - Ye - 1998

Documents on the same site (http://www.csom.umn.edu/WWWPages/Faculty/WLi/): More
Optimal Designs Using CP Algorithms - Li (Correct)
An Integrated Method of Parameter Design and Tolerance Design - William Li (Correct)
Optimal Symmetric Latin Hypercube Designs - William Li (Correct)

Journal of Statistical Planning and Inference
Volume 90, Issue 1 , 1 September 2000, Pages 145-159

doi:10.1016/S0378-3758(00)00105-1 Cite or Link Using DOI

Algorithmic construction of optimal symmetric Latin hypercube designs

Kenny Q. Ye[a], **William Li**[b] **and Agus Sudjianto**[c]

[a] Department of Applied Mathematics and Statistics, SUNY at Stony Brook, Stony Brook, NY 11794-3600, USA

[b] Department of Operations and Management Science, University of Minnesota, Minneapolis, MN 55455, USA

[c] Reliability Methods Department, Ford Motor Company, Dearborn, MI 48121, USA

Received 13 July 1998; revised 16 January 2000; accepted 11 February 2000. Available online 31 July 2000.

Abstract

We propose symmetric Latin hypercubes for designs of computer experiment. The goal is to offer a compromise between computing effort and design optimality. The proposed class of designs has some advantages over the regular Latin hypercube design with respect to criteria such as entropy and the minimum intersite distance. An exchange algorithm is proposed for constructing optimal symmetric Latin hypercube designs. This algorithm is compared with two existing algorithms by Park (1994. J. Statist. Plann. Inference 39, 95–111) and Morris and Mitchell (1995. J. Statist. Plann. Inference 43, 381–402). Some examples, including a real case study in the automotive industry, are used to illustrate the performance of the new designs and the algorithms.

Author Keywords: Computer experiment; Maximum entropy design; Maximin design

Mathematical subject codes: 62K05

Corresponding author

Journal of Statistical Planning and Inference
Volume 90, Issue 1 , 1 September 2000, Pages 145-159

Using response surfaces to improve the search for satisfactory behavior in system dynamics models

Reid Bailey, Bert Bras and Janet K. Allen*

Janet K. Allen is a Senior Research Scientist in Georgia Tech's Woodruff School of Mechanical Engineering. She is interested in design–manufacture of evolving open systems, design in an uncertain environment and modeling design objectives. Dr Allen received her PhD from the University of California at Berkeley in 1973 and her SB from the Massachusetts Institute of Technology.

Bert Bras is an Associate Professor at the George W. Woodruff School of Mechanical Engineering at Georgia Tech. He obtained his MS degree in Mechanical Engineering from the University of Twente, The Netherlands in 1987. After working at the Maritime Research Institute Netherlands, he obtained his PhD in Operations Research from the University of Houston in 1992. His research focus is on environmentally conscious design and manufacturing, design for de- and remanufacture, activity-based costing and life-cycle assessment, and industrial ecology.

Abstract

Traditional system dynamics studies rely heavily upon heuristics and experience. Nevertheless, mathematical exploration techniques have been introduced as important elements for a successful study. We argue that the role of optimization in system dynamics studies is not to replace experience-based knowledge, but instead to augment, facilitate, and expand the heuristic exploration of a model. Accordingly, our approach involves narrowing the design space (using response surfaces) and the subsequent direct investigation of the simulation model (using heuristics). Response surfaces have received considerable attention in optimization because of their capability to replace complex models with analytic equations, thereby increasing computational efficiency. However, doubts exist as to the usefulness of a response-surface approximation of an approximation of reality (i.e., a system dynamics model). We demonstrate the usefulness of response surfaces in system dynamics studies with a case study involving a high-level model of an industrial ecosystem; our intent in using response surfaces is not to replace the simulation models with analytic equations, but instead to direct attention to regions within the design space of the original simulation with the most desirable performance. Recommended changes to a system are based directly on the simulation model, not on response surfaces, avoiding the added level of approximation inherent in response surfaces. The primary focus of the article is on the *concept exploration approach*, which is presented first. The case study towards the end is offered as supporting evidence. Copyright © 2000 John Wiley & Sons, Ltd.

Syst. Dyn. Rev. **16**, 75–90, (2000)

Motivation

System dynamics as a philosophy and feedback control modeling as an implementation of this philosophy are powerful in helping humans understand complex systems. The approaches and methods associated with system dynamics have traditionally been heavily reliant upon the heuristic exploration of possible changes to a system. Nevertheless, optimization techniques have been introduced and used in system dynamics studies dating back to the early 1980s. Whereas some practitioners may have seen a future in which heuristics are completely abandoned in system dynamics studies, most of those exploring the role of optimization emphasized that mathematically rigorous optimization techniques should be *coupled* with heuristics to form a powerful approach for model improvement. Along these lines, our approach is a combination of mathematical tools and heuristics.

*Corresponding Author
Systems Realization Laboratory, G.W. Woodruff School of Mechanical Engineering, Georgia Institute of Technology, Atlanta, Georgia 30332-0405, USA.

System Dynamics Review Vol. 16, No. 2, (Summer 2000): 75–90

Received April 1998
Accepted November 1999

Reid Bailey is a doctoral student in Georgia Tech's Woodruff School of Mechanical Engineering. Reid received his MSME from Georgia Tech in 1997 and his BSE from Duke University in 1995. He is interested in environmental issues in design, multilevel decision-making in complex systems and simulation.

In particular, the mathematical tools we use are design of experiments, response surfaces, and a multiobjective decision model. In using response surfaces, a simulation model is approximated as an analytic equation in which the relevant responses are functions of significant control variables. Response surface techniques do not account for the fact that system dynamics models typically are *rough* approximations of reality—more so than a model that is analytic or based on extensive historical data. While system dynamics models can be constructed of systems that are analytic or based on extensive historical data, a strength of system dynamics lies in the ability to approximate the behavior and structure of poorly understood systems. The usefulness of a response surface approximation of an already highly approximated system dynamics model is questionable. Hence, our approach applies response surfaces only to guide the search for an improved system; the final recommendations, however, are based directly on the simulation model.

Additionally, our approach to 'optimization' does not involve finding one 'optimal' solution, but instead finding a range of *satisficing* solutions. A satisficing solution is one that is 'good enough,' not necessarily optimal (Simon 1996). A major reason for searching for satisficing solutions is that the optimal conditions in a model are most likely not optimal conditions in the real world, especially when modeling poorly understood systems. Therefore, by finding a range of solutions, more play between the model and reality can be accounted for while still achieving the desired responses. We begin by introducing previous work on the use of optimization in system dynamics studies.

Optimization and system dynamics modeling

The inclusion of mathematical tools in the process of improving the behavior of system dynamics models has been examined by several people from varying perspectives. Multiple authors have worked in the area of optimizing the parameters of a model. An excellent starting point to learn about applying optimization to system dynamics models is clearly presented by Wolstenholm (1990) and Wolstenholm and Al-Alusi (1987). These authors do refer to their approach as 'heuristic optimization,' but it is unclear how heuristics influence their optimization process beyond the selection of objective functions and parameters to adjust. Keloharju and Wolstenholm (1989) emphasize the increased efficiency with which a parameter space can be explored using optimization techniques when compared to purely heuristic techniques. They do, however, maintain that human skill remains important in the formulation of the model. Interestingly, they find that several control variable combinations

lead to very similar responses, an idea that supports the search for ranges of solutions introduced in our paper. Coyle's (1985) work is noteworthy in that he includes structural changes to a model through carefully formulating these structural changes with parameters. He, too, emphasizes the role of optimization as a tool *complementary* to the traditional heuristic approaches of system dynamic studies.

Kleijnen (1995) includes design of experiments and response-surface methodology in his approach to optimizing the parameters of a model. Kleijnen introduces these tools as being 'objective, efficient, and effective' and applies them in a moderately standard manner. As in our paper, Kleijnen mixes heuristics and optimization to locate parameter values that produce good behaviors in the model. The difference between our approach and Kleijnen's is that he narrows the design space with heuristics and applies response surfaces to optimize within this smaller region, whereas we apply response surfaces and optimization to narrow the design space and heuristics to locate a *region* of good behavior near the optimized solution. In addition, Kleijnen uses response surfaces to locate the final solution (and consequently checks the solution with the simulation model), while we advocate completely avoiding the added layer of approximation embedded in response surfaces when determining the final solution.

For those not familiar with regression and design of experiments, Rotmans and Vrieze (1990) provide a straightforward review of these techniques, applying them to a simulation model. Rotmans and Vrieze prefer the use of a central composite design, or CCD (which is also used in this paper), when applying design of experiments techniques to their simulation model. As an additional source, the use of orthogonal arrays versus Latin hypercubes in the sensitivity analysis of system dynamics models is discussed by Clemson *et al.* (1995). They point to response surfaces as being more appropriate when 'all parameter interactions need to be studied', which is often the case in optimization.

One final class of approaches for enhancing the analysis and improvement of a system dynamics model is that of modal control, in which eigenvalues of the motion equations are used to synthesize new policy options (Macedo 1989). The main strength of using modal control theory is that new policy structures can be generated mathematically, as opposed to the heuristic methods of Coyle, Kleijnen, and Keloharju. Drawbacks of modal control theory include the necessary linearization of the model, the amount of computation, and the design of realistic policies from the synthetically generated policies (Mohapatra and Sharma 1985). Macedo has introduced a mixed approach in which modal control and traditional optimization are sequentially applied in the improvement of a model. Though we do not focus on Macedo's work, his method appears to be promising.

The approach taken in this paper is similar to that of Keloharju in that the optimization of parameters is the primary focus. However, the implementation of design of experiments and response surface techniques draws upon the work of Kleijnen. A hidden assumption of Kleijnen, that a response-surface approximation of a simulation model (which is an approximation of reality) can be used to determine realistic options for parameter settings, is addressed and handled in this paper. Additionally, our focus is not on determining one 'optimal' solution set of parameter settings, but instead on determining a range of solutions in which system behavior is acceptable. It is these two points of contrast to earlier research that form the contributions of this work.

Frame of reference: robust concept exploration

In recent years, design of experiments and response surfaces have gained in popularity as tools for aiding the exploration of concepts (Box and Draper 1987; Khuri and Cornell 1987; Myers and Montgomery 1995). The basic notion is to relate responses to controllable variables through analytic equations that are constructed through statistical methods. These analytic equations allow for quicker exploration of multiple concepts than do the simulation codes on which they are built.

The robust concept exploration method (RCEM) is a formalization of the steps followed in using response surfaces in the design of a system (Chen *et al.*, 1996). The RCEM is composed of four steps, starting with the identification of design parameters (Step 1). Next, screening experiments are conducted, using design of experiments techniques to reduce the problem size (Step 2). Following the screening experiments, response surfaces are created to map significant design parameters to response behaviors (Step 3). Finally, a multiobjective decision support problem (the compromise Decision Support Problem, or DSP) is formulated using the response surfaces and solved to facilitate concept exploration and the identification of top-level design specifications (Step 4).

The flow diagram in Figure 1 shows the flow of information during the RCEM process. Important to note is that final solutions are based directly on the response surface approximations of the original simulation code.

When the simulation code is based on concrete relationships (such as those from physics or thermodynamics) or on extensive historical data, using response-surface approximations to generate final solutions has been shown to be effective (Koch *et al.* 1996; Myers *et al.* 1989; Simpson *et al.* 1997). From a system dynamics perspective, however, models commonly are rougher

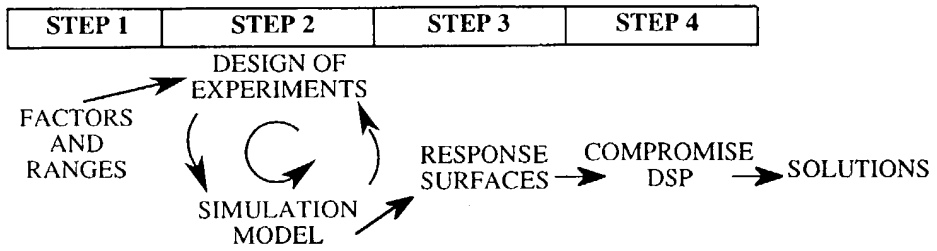

Fig. 1. Robust concept exploration method

approximations of reality—not based on concrete relationships, but instead on human interactions, aggregated trends, and less precise information. The usefulness of a response-surface approximation of a highly approximated simulation model has not been directly addressed in the literature. Our approach to using response surfaces with system dynamics emerges from this dilemma.

Our approach

A response-surface approximation of a simulation model that is already an approximation of reality leads to *two layers of approximation* between the solutions and the real world. If possible, there is a need to eliminate one level of approximation. A way to remove one level of approximation is by adding a loop to the baseline RCEM. With this approach, shown in Figure 2, response surfaces and a compromise DSP are used as *attention-directing* tools, forming an information feedback loop (Bailey 1997).

Instead of taking the solutions directly from a compromise DSP, the results

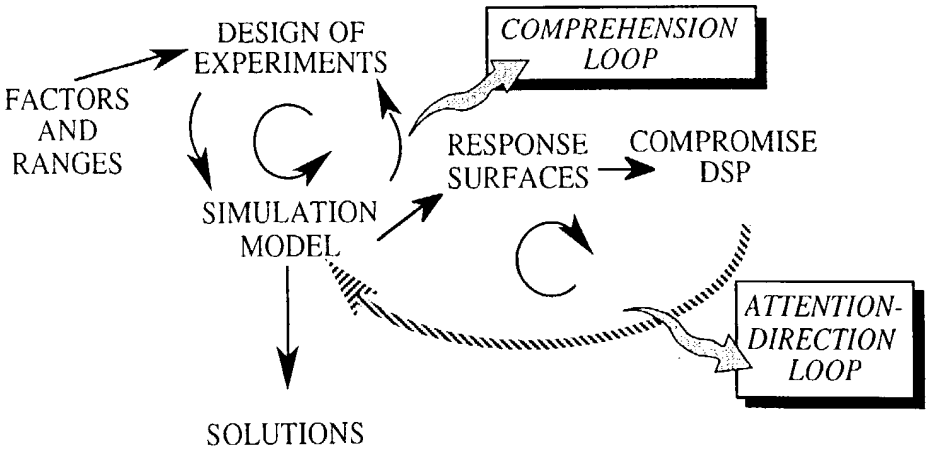

Fig. 2. Augmented RCEM

from a compromise DSP (which are based on response-surface approximations of the simulation model) are fed back to the simulation model. The regions near the compromise DSP solutions are explored with the simulation model in search of a range of acceptable solutions. The response surfaces are still extremely useful in identifying a promising region of the design space, but not in identifying final solutions. The process of creating the response surfaces is termed the *comprehension loop*, because the relationships between responses and control variables are identified and *learned* in this loop.

The added step in the augmented RCEM (shown in Figure 2) involves returning to the simulation program and using the solutions from the compromise DSP formulations as guides to explore the design space. Ranges of control variables for further exploration are determined with two heuristic principles (Bailey 1997):

1. If a compromise DSP solution for a control variable is not on a bound, then two points—one to either side of the compromise DSP solution—are selected for further exploration.
2. If a compromise DSP solution for a control variable is on a bound, then instead of exploring one point on each size of the compromise DSP solution, two points are explored on the side of the compromise DSP solution within the bound.

The points used for further analysis (called *exploration points*), therefore, surround the compromise DSP solution in orthogonal directions. Values for the deviation variables (i.e., the deviations from the desired responses) are found at the exploration points with the original simulation program (*not* the response-surface models) and are then compared to the compromise DSP, or *baseline*, solution. Consider the example shown in Figure 3, in which, for simplicity, a two-dimensional design space is explored (in the case study, the design space is ten-dimensional).

The compromise DSP solution is found and input into Step 1. Points orthogonal to the compromise DSP solution are found for each of the two variables in Step 2. In Step 3, the *deviation variables* at each one of the solution exploration points are determined with the simulation model (these deviations are shown in Figure 3 as vertical columns). Finally, in Step 4, these deviation variables are compared to the deviation variables for the compromise DSP solution. If the deviation variables are within a certain percentage of the compromise DSP solution, then that point is included in the solution range. A designer must determine the allowable amount of fluctuation for each deviation variable (in our case study, a ten percent fluctuation is allowed). If the responses

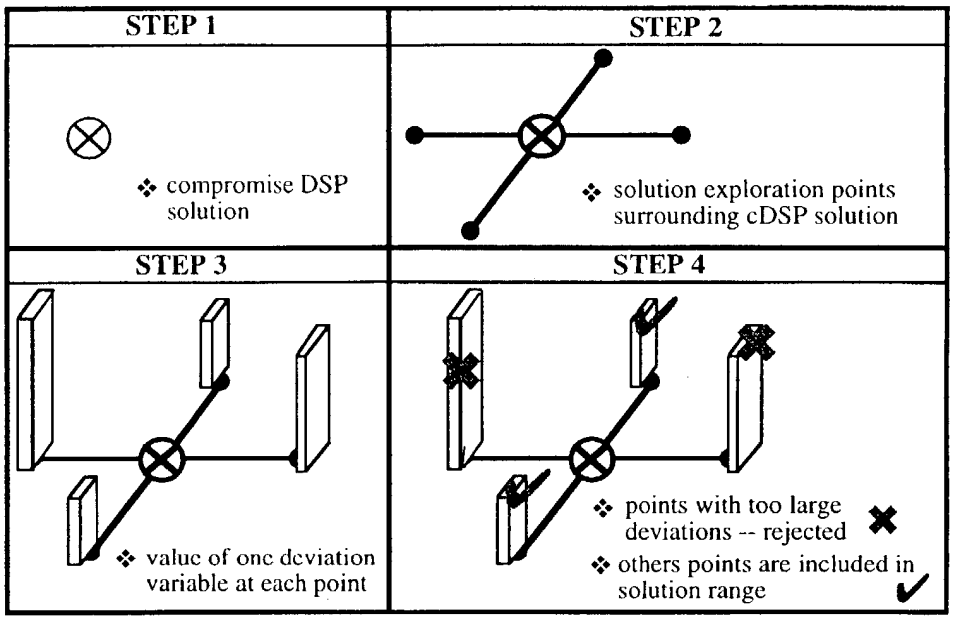

Fig. 3. Solution exploration process

are for a very precise system (e.g., a laser used in surgery), a smaller percentage fluctuation is appropriate.

We discuss two main points concerning the solution exploration strategy:

1. The exploration points are determined heuristically.
2. The solution ranges are based on main effects (Bailey, 1997).

The heuristic determination of the exploration points leads to more flexibility in the solution exploration scheme. A designer has considerable knowledge concerning the sensitivity of the control variables after formulating a decision support problem, developing a simulation model, building response surfaces, and analyzing results from compromise DSPs. Setting concrete rules for determination of the exploration points would limit the process more than enhance it.

The sole use of main effects (i.e., first-order effects) in developing ranged sets of solutions is based on the premise that the interactions and second-order effects do not have much influence on the behavior of the system when small perturbations are made in the values of the control variables. Large changes in the values of the control variables are not made in the solution exploration process, because interaction effects can begin to influence the system behavior more strongly. Quadratic effects, however, *are* modeled by the response surfaces earlier in the concept exploration process. This is necessary to obtain

response surfaces that approximate the behavior of the simulation model accurately enough to search the entire design space.

A benefit of exploring the regions surrounding each control variable is the development of a ranged set of solutions, as opposed to a singular, point solution. With the augmentation made to the RCEM, design freedom is maintained while design knowledge is increased with the solution exploration process. The use of response surfaces and the compromise DSP as *attention-directing* tools (instead of solution-determining tools) is a significant step to maintaining design freedom while gaining information about the system. Instead of using these tools to narrow the design space to one final solution, response surfaces and the compromise DSP are used to expand the information about the design space, which is narrowed to a final range of solutions after an additional step.

The simulation slot in the augmented RCEM (see Figure 2) is where domain-dependent information enters the concept exploration process. The simulation model presented in this paper is of a system of industries in Kalundborg, Denmark.

Case study: a system of industries in Kalundborg, Denmark

Over the past 30 years in Kalundborg, Denmark, a system of industries has developed in which by-products and wastes from one company are used by other companies in the system. For instance, industrial gypsum collects in the scrubbers at Asnæs, a coal and gas burning power plant. This industrial gypsum is sold to Gyproc, where it is used instead of mined gypsum to make wallboard. A series of such interactions exists in Kalundborg and is described in more detail in Bailey *et al.* (1997), Ehrenfeld and Gertler (1997) and Frenay (1995).

A simulation model of a section of Kalundborg (including the industries with the most interactions with other industries) was built, validated, and used to generate data in STELLA (STELLA is described in Chichakly *et al.* 1994). While the model and its validation are described in detail by Bailey (1997), only a brief description is presented here. In Figure 4, a simplified version of the flow diagram of the simulation model is shown.

There are three primary accumulations (steam, gas, and wallboard) that are each individually part of balancing loops. The targets for gas and wallboard are exogenous variables, while the target for the steam accumulation is set by current usage of steam (steam cannot be inventoried indefinitely—it is perishable—so current accumulation levels must be based on steam usage). The interactions among the three companies connect the three balancing loops. For

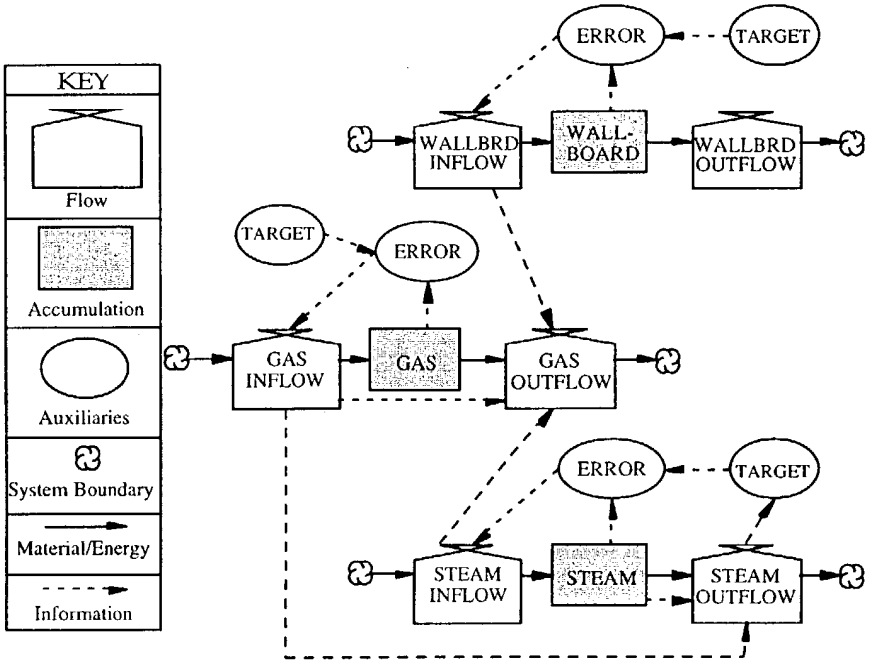

Fig. 4. Simplified flow diagram of Kalundborg model

instance, since steam is used in the purification of gas, an information arrow connects the gas inflow to the steam outflow.

The purpose of the concept exploration process is to design a system that is robust to a disturbance in the system. The primary responses used to evaluate system behaviour are based on the three accumulations, while the controllable variables deal with how the accumulations respond to a disturbance. For the purposes of this paper, only one disturbance—namely, a step change in steam demand—is presented. The responses are selected to characterize the rate of recovery, mean value, and standard deviations of the accumulations as they attempt to reach the stepped target value. Additionally, the total amount of waste generated in the system is calculated and used as a response. Altogether, there are eight responses used to characterize the robustness of the system and the waste generated.

The three types of control factors occur several times within the model, resulting in a total of ten control variables. These delay times, averaging times, and correction times each have physical significance in the actual system:

- *Delay time*: time between the occurrence of a discrepancy between the goal and actual level of an accumulation and the occurrence of first reaction.

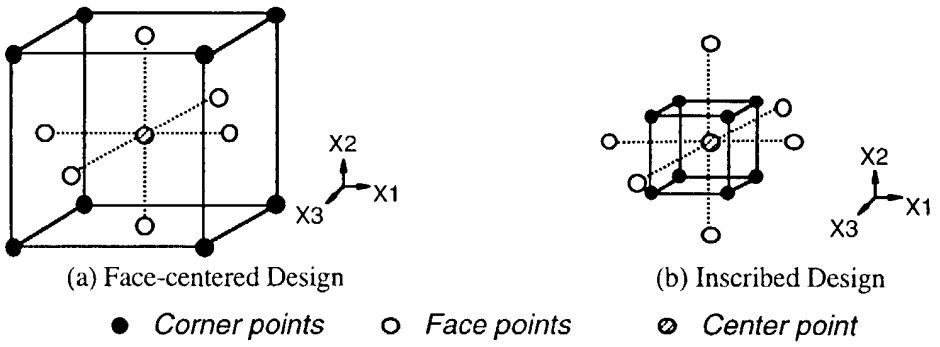

Fig. 5.(a) Face-centered and (b) inscribed central composite designs

- *Averaging time*: time over which flows are averaged; averages of production flows are used in determining present production rates.
- *Correction time*: time over which a discrepancy between the goal and actual levels of an accumulation is intended to be corrected.

All of the decisions in the simulation model relate to changing the production rates of steam, gas or wallboard to maintain the desired accumulation levels. Additionally, the decisions are parametric in nature; they do not involve structural changes in the model. The ten control variables are related mathematically to the eight responses through response surfaces. The basic structure of the response surface formulation is presented in the next section.

Design of experiments and response surface formulation

To develop quadratic response surfaces, experiments must be conducted with the simulation model to determine the effects of each control variable and the interactions between control variables. Among the various types of experimental designs for fitting such response surface models, the central composite design (CCD) is probably the most widely used (Montgomery 1991). For each problem, a slightly different strategy will need to be used to fit response surfaces. The strategy we used is outlined here and is described in more detail by Bailey (1997).

First, a face-centered CCD is used to relate the ten control variables to the eight responses and screen off insignificant factors. A three-factor face-centered CCD is shown in Figure 5(a) with experimental points at the corners (i.e., the bounds for the three factors), the centers of the faces, and in the center of the entire space. In these screening experiments, we found that only four of the eight responses are affected by the step change in steam demand disturbance.

Additionally, the number of significant control variables is reduced from ten to seven.

With the problem reduced in size, a second experiment, using an inscribed CCD, is performed. A three-factor inscribed CCD, shown in Figure 5(b), uses the same number of points as a face-centered design. The difference is that the corner points are moved from the extreme bounds inward, while the face-centered points remain in the same positions. The response surfaces obtained from this second round of experiments are used in the compromise DSP to search for promising regions in the design space (as defined by the values of the control variables). The response surfaces are judged by the R^2 value (coefficient of determination), which represents the proportion of the total variation in the response than can be accounted for by the quadratic relationship with the control variables (Simpson *et al.* 1997; Walpole and Myers 1989). An R^2 of zero means that none of the variation in response can be accounted for by the response surface, whereas an R^2 of one means that all of the variation is accounted for by the response surface. In our case study, all response surfaces have an R^2 of 0.91 or higher.

Compromise DSP solution

The compromise DSP is the mathematical construct through which the various metrics and tools of the RCEM are integrated. It is a multiobjective decision model, which is a hybrid formulation based on mathematical programming and goal programming used to find a set of system variables that satisfy system constraints while achieving a set of conflicting goals as well as possible (Mistree *et al.* 1993). The system descriptors, namely system and deviation variables, system constraints, system goals, bounds, and the deviation function, are described in detail by Mistree *et al.* (1993). The mathematical formulation for a compromise DSP is as follows:

Given: *an alternative to be improved. Assumptions used to model the domain of interest.*

The system parameters:

n	number of system variables
q	inequality constraints
$p+q$	number of system constraints
m	number of system goals
$g_i(X)$	system constraint function
$f_k(d_i)$	function of deviation variables minimized at priority level k for the preemptive case.

Find: *control variables and deviation variables*

$X_i \quad i=1,\ldots,n; d_i^-, d_i^+ i=1,\ldots,m.$

Satisfy:

System constraints (linear, nonlinear)

$g_i(\mathbf{X})=0; i=1,..,p$

$g_i(\mathbf{X}) \geqslant 0; i=p+1,..,p+q$

System goals (linear, nonlinear)

$A_i(\mathbf{X})+d_i^- - d_i^+ = G_i; i=1,\ldots,m$

Bounds

$X_i^{\min} \leqslant X_i \leqslant X_i^{\max}; i=1,\ldots,n$

$d_i^-, d_i^+ \geqslant 0; i=1,\ldots,m$

$d_i^- \cdot d_i^+ = 0; i=1,\ldots,m$

Minimize:

deviation function

$\mathbf{Z}=[f_1(d_i^-, d_i^+),\ldots,f_k(d_i^-, d_i^+)]$

The response surfaces are used in formulating the goals in the compromise formualtion. The particularization of the compromise DSP for the step change in steam usage is presented by Bailey 1997; Bailey *et al.* 1998.

Solution exploration

Following the procedure outlined in the section above describing our approach, solution exploration points are identified that surround the compromise DSP solution. At each of these points, the four deviation variables are calculated using the simulation model of Kalundborg (not the response surfaces). These values of the deviation variables are then compared to the values of the deviation variables for the compromise DSP solution. In Table 1, to illustrate this process, the solution exploration points and associated deviation values are shown for three of the ten control variables.

A solution exploration point is accepted into the final range of specifications only if none of the four values of the deviation variables increases to more than ten percent of the values of the deviation variables from the compromise DSP. In Table 1, all but two deviation variables for the solution exploration points are within an acceptable range. The two unacceptable points (the steam correction time, STMCT, at 3.5 months and the steam delay, STMDEL, at 1.0 months) are those associated with runs #2 and #4. All exploration points that have values of the deviation variables within ten percent of the compromise DSP values (runs 1, 3, 5 and 6) are included in the final solution ranges.

Table 1. Solution exploration step

	Control variables (months)				Deviation variables			
Run	STMCT	STMDEL	...	ATUSE	d_1	d_2	d_3	d_4
cDSP	3.25	0.5	...	12	0.250	1.85e-3	0.0326	0.945
1	3	0.5	...	12	0.219	1.85e-3	0.0337	0.948
	Percentage difference from cDSP solution				−12.4	0	+3.37	+3.17
2	3.5	0.5	...	12	0.283	1.85e-3	0.0317	0.942
	Percentage difference from cDSP solution				***+13.2***	0	−2.76	−0.317
3	3.25	0.75	...	12	0.263	1.85e-3	0.0341	0.944
	Percentage difference from cDSP solution				+5.2	0	+4.6	−0.106
4	3.25	1.0	...	12	0.276	1.85e-3	0.0356	0.944
	Percentage difference from cDSP solution				***+10.4***	0	+9.20	−0.106
5	3.25	0.5	...	10	0.272	1.90e-3	0.0351	0.952
	Percentage difference from cDSP solution				+8.8	+2.70	+7.67	+0.741
6	3.25	0.5	...	11	0.261	1.87e-3	0.0338	0.948
	Percentage difference from cDSP solution				+4.4	+1.08	+3.68	+0.317

* Control valiables not shown are set to cDSP solution variables for the runs shown
STMCT: steam correction time; STMDEL: steam delay; ATUSE: steam usage averaging time

Table 2. Ranged set of solutions (in months)

	Design space bounds		
Control variable	Upper	Lower	Solution values
Steam correction time	0.5	6.0	3.0–3.25
Steam delay time	0.5	2.0	0.5–0.75
Steam generation averaging time	0.5	12.0	5.0–6.0
Steam usage averaging time	0.5	12.0	10.0–12.0
Gas correction time	0.5	6.0	0.5
Gas delay time	0.5	2.0	0.5–1.0
Gas purification averaging time	0.5	12.0	9.25–11.25
Wallboard correction time	0.5	6.0	4.0–6.0
Wallboard delay time	0.5	2.0	0.5–1.0
Wallboard production averaging time	0.5	12.0	10.0–12.0

The range of solutions obtained for the case study, developed using the solution exploration process, is shown in Table 2. The solutions uniformly encourage the use of short delay times and long averaging times, while correction times display no general trend. Shorter delay times are representative of situations in which decisions are made with more current information. Interesting to note is that the solution exploration process has provided a range of solutions that are satisficing (the delay times do not have to be as short as possible to achieve desirable responses). Long averaging times are characteristic of systems that react to disturbances gradually (not abruptly). Therefore, a long-

term approach to handling disturbances in the Kalundborg model results in better performance.

The verification of these results is not shown here; however, it may be found in its entirety in Bailey (1997) and in an abbreviated form in Bailey *et al.* (1998). Through the verification process, the solution ranges are shown to produce the desired robust behaviors in the model.

Closure

There is certainly an opportunity for using design of experiments and response surfaces in the improvement process of a model. The transferal of these tools from other disciplines, however, must be done with care—assumptions concerning their use must be examined. In the case where a model is a rough approximation of reality, the use of response surfaces to determine the final solution is questionable since these response surfaces are further approximations. Nevertheless, response surfaces are powerful and have potential for increasing the efficiency and effectiveness of system dynamics studies. Our approach uses response surfaces to direct attention to a particular location in the design space, and then—returning to the simulation model—develops a range of solutions that will lead to the desired behaviors. The particular implementation of this approach, based on the robust concept exploration method, is outlined in this paper and explored through an example.

One potential drawback to this approach is that the added step—returning to the simulation model—will decrease the efficiency of the process. While some efficiency is lost, the total number of simulation runs added is only twice the number of control parameters—in our example, 20 runs. Including all of the data entry and analysis, the added step required roughly two hours to perform for the Kalundborg example. The benefits of the solution exploration step—using optimization to find a *range of solutions* that are *based directly on the simulation model*—outweigh this slight loss in efficiency.

Approaches such as the one presented here, in which heuristics and mathematical tools are combined in a system dynamics study, are developing as important tools in the improvement of model behavior. For an experienced person, the emphasis might lean towards heuristics, whereas for a novice the mathematical tools can help that person gain experience while reducing the number of major mistakes made along the way. Each component, the heuristic and the mathematical, is an important element to a successful study.

Acknowledgements

Reid Bailey has been supported by a Department of Energy Integrated Manufacturing Fellowship and by NSF grants DMI-96-12327 and DMI 96-25787. We gratefully acknowledge the Engineering Research Program of the Office of Basic Energy Sciences at the Department of Energy and NSF for their support. The cost of computer time has been underwritten by the Systems Realization Laboratory at the Georgia Institute of Technology.

References

Bailey R. 1997. *Designing Robust Industrial Ecosystems: A Systems Approach*, MS Thesis, G.W. Woodruff School of Mechanical Engineering, Georgia Institute of Technology, Atlanta, Georgia.

Bailey R, Allen JK, Bras B, Mistree F. 1997. A system level approach to the design of an industrial ecosystem. *Advances in Design Automation*, Vol. DETC/DAC-3962. ASME: Sacramento, CA.

Bailey R, Bras B, Allen JK. 1999. Using robust concept exploration and system dynamics models in the design of complex systems. *Engineering Optimization* **32**(1): 33–58.

Box GEP, Draper NR. 1987. *Empirical Model-Building and Response Surfaces*. Wiley: New York.

Chen W, Allen JK, Mavris DN, Mistree F. 1996. A concept exploration method for determining robust top-level specifications. *Engineering Optimization*, **26**: 137–158.

Chichakly K, Gass J, Newcomb M, Pease J, Richmond K. 1994. *STELLA II*. High Performance Systems: Hanover, NH.

Clemson B, Tang Y, Pyne J, Unal R. 1995. Efficient methods for sensitivity analysis. *System Dynamics Review*, **11**(1): 31–49.

Coyle RG. 1985. The use of optimization methods for policy design in a system dynamics model, *System Dynamics Review*, **1**(1): 81–91.

Ehrenfeld J, Gertler N. 1997. Industrial ecology in practice: the evolution of interdependence at Kalundborg. *Journal of Industrial Ecology*, **1**(1): 67–79.

Frenay R. 1995. Biorealism, reading nature's blueprints. *Audubon*, 70–79.

Keloharju R, Wolstenholme EF. 1989. A case study in system dynamics optimization. *Journal of Operational Research Society*, **40**(3): 221–230.

Khuri A, Cornell JA. 1987. *Response Surfaces: Design and Analysis*. Marcel Dekker: New York.

Kleijnen JPC. 1995. Sensitivity analysis and optimization of system dynamics models: regression analysis and statistical design of experiments. *System Dynamics Review*, **11**(4): 275–288.

Koch PN, Barlow A, Allen JK, Mistree F. 1996. Configuring turbine propulsion systems using robust concept exploration. *Advances in Design Automation*, Dutta D (ed.), Vol. ASME 96-DETC/DAC-1090. ASME: Irvine, CA.

Macedo J. 1989. A reference approach for policy optimization in system dynamics models. *System Dynamics Review*, **5**(2): 148–175.

Mistree F, Hughes OF, Bras BA. 1993. The compromise decision support problem and the adaptive linear programming algorithm. *Structural Optimization: Status and Promise*, Kamat MP (ed.). AIAA: Washington, DC; 247–286.

Mohapatra PKJ, Sharma SK. 1985. Synthetic design of policy decisions in system dynamics models: a modal control theoretical approach. *System Dynamics Review*, **1**(1): 63–80.

Montgomery DC. 1991. *Design and Analysis of Experiments*. Wiley: New York.

Myers RH, Khuri AI, Carter WH. 1989. Response surface methodology: 1966–1988. *Technometrics*, **31**(2): 137–157.

Myers RH, Montgomery DC. 1995. *Response Surface Methodology: Process and Product Optimization Using Designed Experiments*. Wiley: New York.

Rotmans J, Vrieze OJ. 1990. Metamodelling and experimental design: case study of the greenhouse effect. *European Journal of Operational Research*, **47**: 317–329.

Simon HA. 1996. *The Sciences of the Artificial*, Third edn. MIT Press: Cambridge, MA.

Simpson TW, Peplinski J, Koch PN, Allen JK. 1997. On the use of statistics in design and the implications for deterministic computer experiments. *Design Theory and Methodology*, Vol. DETC/DTM-3881. ASME: Sacramento, CA.

Walpole RE, Myers RH. 1989. *Probability and Statistics for Engineers and Scientists*, Fourth edn. MacMillan: New York.

Wolstenholme EF. 1990. *System Enquiry: A System Dynamics Approach*. Wiley: New York.

Wolstenholme EF, Al-Alusi A. 1987. System dynamics and heuristic optimisation in defence analysis. *System Dynamics Review*, **3**(2): 102–115.

Using Statistically Designed Experiments to Improve Reliability and to Achieve Robust Reliability

Michael Hamada
University of Waterloo, Waterloo

Key Words — **Bayes, Box-Behnken design, Censored data, Control & noise factors, Experimentation, Fractional & full factorial, Maximum likelihood, Mixed-level design, Plackett-Burman design, Regression model, Robust design, Response surface design, Sequential approach, Weibull.**

Summary & Conclusions — **This tutorial explains statistically designed experiments which provide a proactive means to improve reliability as advocated by Genichi Taguchi. That is, by systematic experimentation, the important parameters (factors) affecting reliability can be identified along with parameter values that yield reliability gains. In addition to improving reliability, Taguchi's robust design can be used to achieve robust reliability; that is, to make a process or product reliability insensitive to factors which are hard or impossible to control. Robust design is also implemented using statistically designed experiments. This paper presents classes of experimental plans for reliability improvement and robust reliability. An important feature of the reliability data collected from such experiments is censoring which occurs when some of the experimental units have not failed by the end of the experiment. Consequently, the analysis methodology must account for these censored data which are likely to occur in light of the ever increasing reliability of today's products. Several appropriate methods are discussed briefly. These experimental plans and analysis methods are illustrated using three documented experiments which improved fluorescent lamp and industrial thermostat reliability and which achieved robust reliability for night-vision goggles.**

1. INTRODUCTION

Statistically designed experiments have been used extensively for estimating or demonstrating existing reliability, *eg*, most of the examples in Nelson [15]. Until recently, designed experiments appear to have seldom been used to improve reliability by identifying the important parameters (factors) affecting reliability out of many potentially important ones. For example, Taguchi [25 - 27] advocated using designed experiments as a proactive means for improving reliability and provided examples of improving clutch springs and fluorescent lamps. Taguchi is perhaps best known for robust-design, whose aim is to make processes/products insensitive to noise factors which are hard or impossible to control. Such products/processes are robust to the noise factors. Examples of noise factors include manufacturing variables that cannot easily be controlled, and environmental conditions in which the product is used. This important paradigm for improving products/processes, which attracted the attention of industry in the 1980s [11] can also be applied to reliability. To ensure good stability and adequate reliability, Taguchi [25: page 149] recommended that noise factors be considered in any experiment to improve reliability when it is practical to do so.

Since the early 1980s, several experiments for improving reliability have been documented. Specht [23] reported the improvement of heat-exchanger reliability in a commercial heating system; Montmarquet [14] discussed the improvement of drill-bit reliability in a multilayer printed circuit board drilling operation; and Reed [19] presented the improvement of night-vision goggle reliability. From an early application of Taguchi's methodology at AT&T, Phadke [16] discussed the improvement of router-bit reliability in a printed circuit board cutting operation. Condra [4] gave several examples from the electronics industry. Taguchi's robust design philosophy figures prominently in Condra's book. Recently, Bullington, Lovin, Miller, Woodall [3] reported on the improved reliability of industrial thermostats.

This tutorial illustrates how statistically designed experiments can be used to improve reliability and to achieve robust reliability. Classes of experimental plans for improving reliability and achieving robust reliability are discussed with examples in sections 2 & 3, respectively. Section 4 briefly discusses analysis methodology for extracting the information from the experimental data. The methodology must account for censoring, an important feature of such experiments, which occurs when some of the experimental units have not failed by the end of the experiment; this type of censoring produces type-I right-censored data. Other types of censoring typically arise in such experiments. Where units cannot be monitored continuously, units must be inspected periodically until failure. Periodic inspection produces left-censored data for units failing before the first inspection and interval-censored data otherwise. Appropriate analysis techniques which handle censored data are illustrated in sections 5 - 7 which analyze experiments to improve fluorescent-lamp and thermostat reliability and to achieve robust reliability of night-vision goggles.

Acronyms[1]

ML maximum likelihood
MLE ML estimate.

Assumptions

1. Lifetimes are s-independent and either Weibull or lognormally distributed.
2. Some lifetimes are censored. ◄

[1]The singular & plural of an acronym are always spelled the same.

Notation

2^{k-p}, 3^{k-p} [2-level, 3-level] fractional factorial, k factors
Y_i response i
T_i lifetime i
x_i vector i of covariate values associated with Y_i
β vector of regression parameters or effects
σ scale parameter
ϵ_i error i associated with Y_i
x_{control} vector of control-factor levels, also covariate values in the regression model involving only the control factors
x_{noise} vector of noise-factor levels, also covariate values in the regression model involving only the noise factors
$x_{\text{control}\times\text{noise}}$ covariate values in the regression model corresponding to control-noise interactions
$\{i \in \text{CENS}\}$, $\{i \in \text{FAIL}\}$ set of [censored, failure] data
$\mathcal{L}(\cdot)$ likelihood function
A_j level j of factor A.

Other standard notation is given in "Information for Readers & Authors" at the rear of each issue.

2. EXPERIMENTS FOR IMPROVING RELIABILITY

While there are potentially many factors (parameters) that affect reliability, some factors are more important, *viz*, have a bigger impact on reliability as the values of these factors are changed. These important factors can be identified empirically through experimentation which involves making deliberate changes in the factor values (levels) and observing the resulting reliability. Besides identifying the important factors, levels for these factors that yield reliability gains can be recommended. Statistically designed experiments provide a systematic & efficient plan of experimentation to achieve these goals so that several factors can be studied simultaneously. Designed experiments, which have successfully been used to improve other quality characteristics can also be used to improve reliability.

2.1 Full Factorial Designs

Some terminology is helpful in describing the plans. The plan of experimentation is the *experimental design*, or simply *design*. The experimental design consists of a list of runs (a run is a combination of levels at which the factors in the experiment are set). The number of runs in the experimental design is the run-size. The experiment then involves making units according to the conditions specified by the runs in the experimental design and lifetesting these units to failure. Table 1 gives an experimental design for 3 factors (A, B, C) with each factor being studied at 2 levels (1, 2). Run 1 indicates that units are made with all the factors set at their respective level 1's. This particular design is a '2-level full-factorial' since it consists of all possible combinations of the 2 levels for the 3 factors.

TABLE 1
2-Level Full Factorial Design for 3 Factors

	Factor		
Run	A	B	C
1	1	1	1
2	1	1	2
3	1	2	1
4	1	2	2
5	2	1	1
6	2	1	2
7	2	2	1
8	2	2	2

The run size for a 2-level full-factorial design in k factors is 2^k, which quickly becomes prohibitive for more than 5 factors. Designs with more than 2 levels allow curvature, interactions, or nonlinear effects to be assessed but require more runs. Even for 3 factors, the run size of the full-factorial design is $3^3=27$. Because of their large run sizes, full-factorial designs tend not to be used in an initial experiment unless there are only a few potentially important factors to be studied.

2.2 2-Level Fractional Factorial Design

For the typical industrial situation, many factors need to be studied in a relatively few runs. A sequential approach to experimentation provides one such strategy. An initial experiment using only a few levels (often 2) for each factor is performed to screen out the unimportant factors. A follow-up experiment involving much fewer factors but at more levels can then be used to explore the response-factor relationship in more detail. For the initial experiment, a subset or fraction of the full factorial design (a fractional factorial) can be used. There are two types of 2-level designs:

- geometric (2^{k-p}) designs [2]
- non-geometric designs [17].

The notation for the geometric designs indicate the degree of fractionation, *viz*, a 2^{-p} fraction of a full factorial with run size 2^{k-p}.

Taguchi [26: page 930] provided an example of a 2^{k-p} design which was used to improve the reliability of fluorescent lamps. The experiment studied five 2-level factors (A - E) in 8 runs using a 2^{5-2} design (a quarter of a full factorial) as given in table 2. Eight types of lamps were built — contrasted with a full-factorial design which would have required 32 types. No further details on the factor names and levels were provided. Two lamps were made at each run, and life testing was conducted over 20 days with inspections for failure every 2 days. Seven of the 16 lamps had not failed by the 20-day inspection, which yielded right-censored data.

Bullington *et al* [3] provided an example of a 12-run Plackett-Burman design which was used for an experiment

TABLE 2
Design & Lifetime Data
[Fluorescent-Lamp Experiment]

Factor								
A	B	C	D	E	?	?	Lifetime*	
1	1	1	1	1	1	1	(14,16)	(20,∞)
1	1	2	2	2	1	2	(18,20)	(20,∞)
1	2	1	1	2	2	2	(08,10)	(10,12)
1	2	2	2	1	2	1	(18,20)	(20,∞)
2	1	1	2	1	2	1	(20,∞)	(20,∞)
2	1	2	1	2	2	2	(12,14)	(20,∞)
2	2	1	2	2	1	2	(16,18)	(20,∞)
2	2	2	1	1	1	1	(12,14)	(14,16)

*The lifetime is uncertain. The 2 endpoints of the interval, within which the lifetime lies, are shown in (). An end point of "∞" implies that the item had not failed at the end of the test. The time-unit is *days*.
?=This design could have accomodated 2 more factors in these 2 unused columns.

to improve the reliability of industrial thermostats. Eleven factors (A - K) were studied using the design in table 3 in which 10 thermostats were manufactured at each of the 12 run settings, *viz*, 12 types of thermostats were built. (A full-factorial design would have required $2^{11} = 2048$ types.) These factors were chosen from many across a 14-stage manufacturing process and include: E — beryllium-copper grain size, H — heat treatment, J — power element electroclean, and K — power element plating rinse. Each factor was studied at 2 levels, *eg*, E — grain size of 0.008 & 0.018 inches, and H — 45 & 240 minutes. Table 3 also presents the lifetime data; the experiment was stopped at 7432 k-cycles resulting in 22 right-censored observations at runs 1, 6, 11.

While highly fractionated 2^{k-p} and Plackett-Burman designs are ideally used as screening designs, in practice, the initial experiment might be the only one performed. Consequently, a properly chosen 2^{k-p} design can allow some interactions to be studied. For example in the fluorescent-lamp experiment, besides the factors A - E main effects, the experimenter also thought that the $A \times B$ interaction might be important; this experimental design (table 2) allows the $A \times B$ interaction to be estimated. Factor *main-effects* refers to the additive effects of the factors on reliability. The interaction between two factors indicates the degree of non-additivity of the factor effects; *ie*, if an interaction is present, the effect on reliability of changing the levels of one factor depends on the level of the other factor. Consequently, the presence of an interaction can impact the recommendations of levels for setting the important factors. See [2] for more discussion. For Plackett-Burman designs, [9] showed that some information on interactions can be obtained. Ref [9] proposed a sequential strategy which first identifies the important factor main effects. Then, interactions involving these identified factors are entertained for possible importance.

2.3 Multi-Level Designs

Taguchi [26] often initially uses designs with more than 2 levels. These include the 3^{k-p} designs and mixed-level designs such as the 18-run design which can be used to study one 2-level factor and up to seven 3-level factors. Refs [5, 28] described other mixed-level designs. An example of the use of these designs is the clutch-spring experiment in Taguchi [25: chapter 9] which used a 3^{7-4} design to study 7 factors in 27 runs, a 1/81 fraction of a 3-level full factorial design. Thus, each of the factors was studied at 3 levels. Phadke [16] also used a mixed-level 32-run design to study two 4-level factors and seven 2-level factors in a routing process. One of the 4-level factors was router bit type so that 4 types of bits were studied. Thus, qualitative factors such as the router bit type can be investigated as well as quantitative factors such as temperature.

TABLE 3
Design & Lifetime Data
[Thermostat Experiment]

Design											Ordered Lifetime Data (k-cycles)								
A	B	C	D	E	F	G	H	I	J	K									
1	1	1	1	1	1	1	1	1	1	1	957	2846	*						
1	1	1	1	1	2	2	2	2	2	2	206	284	296	305	313	343	364	420	422	543
1	1	2	2	2	1	1	1	2	2	2	63	113	129	138	149	153	217	272	311	402
1	2	1	2	2	1	2	2	1	1	2	76	104	113	234	270	364	398	481	517	611
1	2	2	1	2	2	1	2	1	2	1	92	126	245	250	390	390	479	487	533	573
1	2	2	2	1	2	2	1	2	1	1	490	971	1615	6768	*					
2	1	2	2	1	1	2	2	1	2	1	232	326	326	351	372	446	459	590	597	732
2	1	2	1	2	2	2	1	1	1	2	56	71	92	104	126	156	161	167	216	263
2	1	1	2	2	2	1	2	2	1	1	142	142	238	247	310	318	420	482	663	672
2	2	2	1	1	1	1	2	2	1	2	259	266	306	337	347	368	372	426	451	510
2	2	1	2	1	2	1	1	1	2	2	381	420	*							
2	2	1	1	2	1	2	1	2	2	1	56	62	92	104	113	121	164	232	258	731

*Test was stopped at 7342 (right censored); this and subsequent items did not fail.

In contrast with the initial use of multi-level factor designs, the sequential approach to experimentation uses multi-level designs in a follow-up experiment. Box & Draper [1] gave response-surface designs which allow the response-factor relationship to be explored in more detail. For example, an internal document from a North American automobile manufacturer reported the use of an 8-run experiment to screen 7 factors. (The same design given in table 2 consisting of the first 7 columns was used.) Four factors were identified, and were studied further using the 27-run Box-Behnken design in table 4. (-1, 0, +1) denotes the 3 levels for each factor; each of the groups of rows, 1 - 2, 4 - 5, 7 - 8, specify 4 runs since the ± 1 notation means that all combinations of levels 1 & 3 are used. Other response surface designs such as the central-composite designs (with 5 factor-levels) can be employed. See [1] for more details.

TABLE 4
Box-Behnken Response-Surface Design for 4 Factors

Factor			
A	B	C	D
±1	±1	0	0
0	0	±1	±1
0	0	0	0
±1	0	0	±1
0	±1	±1	0
0	0	0	0
±1	0	±1	0
0	±1	0	±1
0	0	0	0

2.4 Analysis of Lifetime Data

The lifetime data from these experimental designs can be analyzed using a parametric regression model such as the lognormal or Weibull. These models have the form [12]:

$$Y_i = \log(T_i) = x_i^T \cdot \beta + \sigma \cdot \epsilon_i,\ i = 1,\ldots,n. \quad (1)$$

For the Weibull model, the errors $\{\epsilon_i\}$ are i.i.d. standard extreme-value r.v., whose Sf is exp(exp(-w)). For the lognormal model, the errors $\{\epsilon_i\}$ are i.i.d. standard s-normal r.v., whose Sf is gaufc(w). These models assume that the factors affect the mean log(lifetime) (through $x^T \cdot \beta$) and do not affect the common scale-parameter σ. In terms of these models, the analysis first seeks to identify the non-zero components of the vector of effects β. The recommended levels for setting the important factors are those levels which maximize the mean log(lifetime) $x^T \cdot \beta$.

2.5 Design *vs* Corresponding Covariates

Table 5 gives the corresponding covariate values for the 2-level 3-factor full-factorial from table 1.

Nomenclature

INT an intercept
A,B,C factor main effects
$A \times B, A \times C, B \times C$ 2-factor interactions.

To obtain the main effects values, [1, 2] in table 1 are replaced by [-1, +1]. Values for the interactions are obtained by multiplying those from the corresponding main effects, *eg*, the $A \times B$ values are obtained by multiplying the A & B main-effect values. Ref [13] discussed covariate values for more than 2 level-factors.

TABLE 5
Covariates for 2-Level 3-Factor Full-Factorial Design

		Values					
Run	INT	A	B	C	$A \times B$	$A \times C$	$B \times C$
1	1	-1	-1	-1	+1	+1	+1
2	1	-1	-1	+1	+1	-1	-1
3	1	-1	+1	-1	-1	+1	-1
4	1	-1	+1	+1	-1	-1	+1
5	1	+1	-1	-1	-1	-1	+1
6	1	+1	-1	+1	-1	+1	-1
7	1	+1	+1	-1	+1	-1	-1
8	1	+1	+1	+1	+1	+1	+1

Section 4 discusses analysis methodology for fitting these models in (1) using censored data. Sections 5 & 6 present analyses of the fluorescent-lamp- and thermostat-experiments respectively.

3. EXPERIMENTS FOR ACHIEVING ROBUST RELIABILITY

Taguchi's robust design is also referred to as parameter-design because its objective is to find levels of engineering parameters (control factors) that yield a robust product/process, *ie*, that make the product/process insensitive to the variation of hard- or impossible-to-control noise factors. Robust design is therefore different from the approach of handling sources of variation by control, which can be costly. For example, the Ina Tile Company was faced with reducing an unacceptable amount of variation in their tile-size caused by an uneven temperature distribution in the kiln [11]. Rather than purchasing an expensive kiln which would have controlled the temperature distribution better, it was found through designed experiments that increasing the lime content in the tile formulation decreased the tile size variation by a factor of 10. In other words, a tile formulation was found that was insensitive to the existing oven's uneven temperature distribution.

Taguchi's tactics for robust design are to specify a criterion for assessing the effect of the noise factors and to estimate it by experimentation. While noise factors are difficult or impractical to control in production or in use, for purposes of the experiment (to learn about the effect of the noise factors), the noise

factors need to be controlled during the experiment. The s-expected loss $L(\cdot)$, a criterion for assessing the effect of the noise factors at a particular combination of control factor levels x_{control}, can be defined for a general loss function $l(\cdot)$ [29]:

$$L(x_{\text{control}}) = \int l(Y(x_{\text{control}}, x_{\text{noise}})) \cdot f(x_{\text{noise}})\, dx_{\text{noise}}. \quad (2)$$

Notation

$l(\cdot)$ loss function
$Y(x_{\text{control}}, x_{\text{noise}})$ response at combination of control factor levels $(x_{\text{control}}, x_{\text{noise}})$
$L(x_{\text{control}})$ s-expected loss at x_{control}
$f(\cdot)$ joint pdf of noise factors.

The objective of robust design is to find a product/process design, x_{control}, with minimum s-expected loss. An appropriate loss function for reliability is discussed in section 3.3.

3.1 Product Array Designs

Taguchi [26] proposed estimating the loss (2) via experimentation according to specialized experimental plans (product arrays). A product array consists of 2 plans or arrays, one for the control factors (control array) and the other for the noise factors (noise array). The product-array design is so named because all the noise-factor combinations specified by the noise array are run with every combination of the control factors specified by the control array.

As an example, consider an experiment to improve the reliability of a night-vision goggle tube sub-assembly [19]. The tube sub-assembly is insulated by a combination of protective coatings which degrades over time and over exposure to humidity & temperature. One goal of the experiment was to make the tube insulation reliability robust to the handling of tube when the coatings are applied. The noise factor, N, had 2 levels (the tube is handled or not). Ten control factors (A - J) were chosen, from many, across a 17-step manufacturing process and include factors related to the tube packaging such as type of coating, primer, and electrical-connection configuration. Each control factor was also studied at 2 levels according to a 12-run Plackett-Burman design, *eg*, 2 types of primer coating (factor B) and 2 types of lead coating material (factor H) were used. Therefore, the product array consisted of a 12-run Plackett-Burman design for the control factor array and a simple noise factor array (1 factor at 2 levels) as displayed in table 6. At each of the 24 control/noise factor combinations, one tube was manufactured and then life tested under a high-temperature cycling and humidity regimen. The tubes were inspected for failure every 2 days for 20 days. The lifetime data in table 6 make some assumptions because it is unclear from [19] whether 16 of the tubes failed between days 18-20 or whether they were still functioning at 20 days. For illustration, they are treated here as still functioning, resulting in right-censored data; 6 of the tubes also failed before the inspection #1 at day 2, yielding left-censored data. The product-array design used in this experiment is a fractional-factorial design which would otherwise have required $2^{11} = 2048$ control and noise factor settings.

TABLE 6
Product-Array Design & Lifetime Data
[Goggle Experiment]

Control Array										Noise Array, N	
A	B	C	D	E	F	G	H	I	J	1	2
1	1	1	1	1	1	1	1	1	1	(0,2)	(20,∞)
1	1	1	1	2	2	2	2	2	2	(20,∞)	(20,∞)
1	1	2	2	1	1	1	2	2	2	(0,2)	(0,2)
1	2	1	2	1	2	2	1	1	2	(7,9)	(20,∞)
1	2	2	1	2	1	2	1	2	1	(20,∞)	(20,∞)
1	2	2	2	2	2	1	2	1	1	(20,∞)	(7,9)
2	1	2	2	1	2	2	1	2	1	(20,∞)	(20,∞)
2	1	2	1	2	2	1	1	1	2	(20,∞)	(0,2)
2	1	1	2	2	1	2	2	1	1	(0,2)	(0,2)
2	2	2	1	1	1	2	2	1	2	(20,∞)	(20,∞)
2	2	1	2	2	1	1	1	2	2	(20,∞)	(20,∞)
2	2	1	1	1	2	1	2	2	1	(20,∞)	(20,∞)

3.2 Analysis of Product Array Design Data

For analyzing the product-array data, Taguchi [26] proposed estimating the s-expected loss $L(x_{\text{control}})$ (2) for each x_{control} specified by the control array from the data obtained by varying the noise factors according to the noise array and then modeling the estimated losses in terms of the control factors. That is, he proposed constructing signal-to-noise ratios from the noise-array data and analyzing them by common methods for designed experiments such as analysis of variance. Ref [21] referred to Taguchi's approach as the loss-model approach. Alternatively, [29] proposed modeling the response Y directly in terms of both the control & noise factors and then evaluating the loss using the estimated response model. Their rationale for the latter (response-model [21]) approach was that it would be more likely to find a simple model for the response than one for the much more complicated estimated s-expected loss. Examples in [21, 29] gave evidence for preferring the response-model approach because it provides additional insight.

For reliability applications, the response-model approach is reasonable because the same parametric regression models in section 2 can be used. An even more compelling reason against using the loss-model approach is that the signal-to-noise ratios cannot handle censored data. The product-array data allow model (1) to be fit consisting of all Control main effects (with possibly some Control×Control interactions), all Control×Noise interactions and all Noise main effects (with possibly some Noise×Noise interactions), where Control & Noise denote specific control & noise factors, respectively. The covariate values have the form:

$$x = (1, x_{\text{control}}, x_{\text{control}\times\text{noise}}, x_{\text{noise}}),$$

for the main effect values,

$(x_{\text{control}}, x_{\text{noise}})$

and corresponding Control×Noise interaction values,

$x_{\text{control}\times\text{noise}}$.

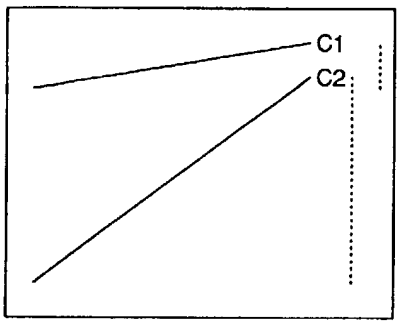

Figure 1. Example Response-Functions Showing Opportunity for Robustness

The Control×Noise interactions are important in achieving robust reliability because the fact that the s-expected loss (2) changes for various control-factor combinations means that these interactions must exist. Figure 1 displays a simplified relationship between a response Y and 1 control-factor C (at 2 levels) and 1 noise-factor N (varied over an interval). The vertical lines in figure 1 show that the effect of the noise factor on the response Y is substantially smaller at control-factor level 1 (C1) as the noise factor varies and hence is more robust at that level. Consequently, robust design exploits the existence of such interactions between control- and noise-factors.

3.3 Choosing Parameter Values (Factor Levels)

Once estimates for the response-model effects have been obtained, the important control-factor levels need to be recommended. For a simple model with few noise factors, recommended levels might be apparent from inspecting the model directly; *ie*, by observing what the important effects. Shoemaker *et al* [21] gave an example, but for complicated models, their approach might be tedious.

An alternative is to specify some meaningful criterion or s-expected loss (2) and use the identified response model (1) to evaluate them, assuming some distribution for the noise factors. In practice, where it is difficult to specify a distribution for the noise factors, the criterion can be evaluated over a sample of noise combinations as specified by a noise array. The same noise array from the experiment need not be used, however. For example, instead of using a fractional-factorial design, the s-expected loss could be evaluated over a full-factorial design. Similarly, the s-expected loss could be evaluated for all possible settings of the control factors, not just those specified by the control array.

For achieving robust reliability, therefore, reliability should depend as little as possible on the noise factors. Also high average reliability is required. Ref [6] considered criteria based on the linear part $(x^T\cdot\beta)$ of model (1), *viz*, the mean log(lifetime) for the lognormal regression model. This paper assesses reliability in terms of the probability of exceeding a certain time T, such as a warranty period. Using model (1), this survival probability is:

$$\text{Sf}\{(\log(T) - x^T\cdot\beta)/\sigma\}. \tag{3}$$

Notation

Sf{} an appropriate survivor function
x $(1, x_{\text{control}}, x_{\text{control}\times\text{noise}}, x_{\text{noise}})$.

Thus, using (3) as the loss function in (2) means that the s-expected loss here is the s-expected reliability. For a given control-array combination, the s-expected reliability can be estimated by evaluating the reliabilities (3) over all the noise-array combinations and then taking their mean. These evaluated reliabilities (over the noise array) represent a sample of reliabilities which can be summarized in other meaningful ways; *eg*, the standard deviation which measures the variability of the reliabilities as the noise factors vary. By taking a worst-case approach, the minimum of the sample can also be compared. Based on these criteria, control-array combinations with large mean, large minimum, and small standard deviation are desirable. Analysis of the goggle experiment in section 7 illustrates the use of these criteria.

3.4 Usual Taguchi Robust Reliability-Design

Most of the robust design applications involve a continuous quality characteristic Y with ideal or target value Y_0 and use the squared error loss function $(Y-Y_0)^2$ in (2). Provided factors can be found which can adjust the mean quality characteristic to target, then robust design amounts to minimizing variability of the quality characteristic caused by the variation of the noise factors. Situations in which reliability is defined in terms of the quality characteristic exceeding a threshold fall into this category. For example, Taguchi often gives the example of an electronic system in which the parameters of the components drift with age. In this example, the control factors levels are nominal values for the component parameters and the noise factors levels are deviations in the component parameters from the nominal values that reflect the drifting. No lifetesting is required here, which is a distinct advantage of analyzing the quality characteristic directly. This paper focuses on situations which do not have such continuous quality characteristics, however, so that lifetesting must be done.

4. ANALYSIS METHODS FOR CENSORED DATA

For model (1), when all the lifetimes are observed (complete sample), the analysis is straightforward using ML estimation [12]. The MLE for (β, σ) are found by maximizing the following likelihoods:

For the Weibull regression model,

$$\mathcal{L}(\beta, \sigma) = \prod_i (1/\sigma)\cdot\exp[[(y_i - x_i^T\cdot\beta]/\sigma] - \exp[(y_i - x_i^T\cdot\beta)/\sigma]].$$

For the lognormal regression model,

$$\mathcal{L}(\beta, \sigma) = \prod_i (1/\sigma)\cdot\text{gaud}[(y_i - x_i^T\cdot\beta)/\sigma].$$

The estimates are compared with their standard errors [12] to identify the important effects. The problems arising from censored data are discussed next.

One method which unwisely continues to be used treats the right-censoring times as actual failure times and then analyzes them by common methods for complete data. (For interval-censored data, an interval endpoint or midpoint might be used.) Although simple, ignoring the censoring can lead to wrong decisions because the unobserved failure times and right-censoring times can differ greatly, depending on the particular factor-level combination. A simulation [8] showed that this method can perform quite poorly by missing some important effects and mis-identifying spurious effects.

ML estimation easily handles both failure and censored (right or left) data. The MLE for (β, σ) are found by maximizing the following likelihoods:
For the Weibull regression model,

$$\mathcal{L}(\beta, \sigma) = \prod_{i\in\text{FAIL}} (1/\sigma)\cdot\exp\{[(y_i - x_i^T\cdot\beta]/\sigma] - \exp[(y_i - x_i^T\cdot\beta)/\sigma]\}\cdot\prod_{i\in\text{CENS}}\exp\{\exp[-(y_i - x_i^T\cdot\beta)/\sigma]\}. \quad (4)$$

For the lognormal regression model,

$$\mathcal{L}(\beta, \sigma) = \prod_{i\in\text{FAIL}} (1/\sigma)\cdot\text{gaud}[(y_i - x_i^T\cdot\beta)/\sigma]\cdot\prod_{i\in\text{CENS}}\text{gaufc}[(y_i - x_i^T\cdot\beta)/\sigma]. \quad (5)$$

Standard errors for the MLE can be obtained [12]. Commercially available software perform these computations, such as SURVIVAL, the SYSTAT survival analysis module [24], or the LIFEREG procedure in SAS [20]. For a censored datum, its contribution to $\mathcal{L}$ is simply the probability of being censored. Similarly for an interval-censored datum (a,b); its contribution to $\mathcal{L}$ is the probability of failing between times a & b.

One problem with ML estimation for censored data is that the MLE might not exist, *ie*, at least one parameter estimate is infinite, so that testing for important effects cannot be done by comparing the MLE with their standard errors. Ref [22] gave necessary & sufficient conditions for the existence of MLE for model (1). In the reliability context, [7] concluded that estimability problems tend to occur for the designs in sections 2 & 3 where the fitted model has nearly the same number of parameters as number of observations.

The estimability problem of the ML approach motivated the Bayes approach [10]. The Bayes approach is reasonable because important factor effects might be anticipated to be large but not infinite. By using appropriate prior distributions, well-behaved posterior distributions can be calculated whose finite modes or medians can be used as estimates of the effects. Also, posterior distributions allow the importance of factorial effects to be assessed without using the asymptotic approximations needed by the ML method; a simple way to assess whether a factor effect is important is to see whether its marginal posterior does not contain zero. Ref [10] considered the lognormal regression model (1) and used the reasonable conjugate prior [18]; the prior pdf is:

$$p(\beta, \sigma) = \sigma^{-k}\cdot\exp[-(\beta - \beta_0)^T\cdot A_0(\beta - \beta_0)/(2\sigma^2)]\cdot\sigma^{-(\nu_0+1)}\cdot\exp[-\nu_0\cdot s_0^2/(2\sigma^2)]. \quad (6)$$

The posterior is proportional to the product of the likelihood (5) and the prior (6) and is relatively simple to obtain numerically using recent advances in Bayes computing. An appropriate choice of the prior parameters $(\nu_0, s_0^2, \beta_0, A_0)$ gives a very diffuse or essentially non-informative prior which means that the results reflect the information provided by the data. The sensitivity of the results based on the current data can be assessed by trying more informative priors [10]. The Bayes approach is illustrated in section 7 in the analysis of the goggle experiment.

5. ANALYSIS OF THE FLUORESCENT-LAMP EXPERIMENT

Consider the fluorescent-lamp experiment in section 2. The experiment studied 5 factors $(A - E)$ using the design in table 2. Besides the 5 main effects $(A - E)$, the experimenters thought that the $A\times B$ interaction might be important. Using ML estimation, a lognormal regression model was fit using the lifetime data in table 2. Table 7 gives the MLE and s-significance levels (p values) for the 5 main effects (A - E) and the $A\times B$ interaction (the intercept is denoted by INT). Based on these results, the main effects D, B, E, A are important (small p values) in the order given. Therefore, only 4 of the 5 factors are important, with A being only marginally important. Since the $A\times B$ interaction is not important, the signs of the important main effects can be used to recommend factor levels. If the effect is negative, then level 1 (whose covariate value is -1) improves reliability, *viz*, the mean log(lifetime). Similarly, for a positive effect, level 2 (whose covariate value is $+1$) improves reliability. Thus the results from table 7 suggest that reliability can be improved at factor levels $A_1B_2D_1E_2$, where the subscript indicates the recommended level.

Table 7
MLE and P Values of Lognormal Regression Model
[Fluorescent Lamp Experiment]

Effect	MLE	P Value
INT	2.939	0.000
A	-0.117	0.059
B	0.201	0.001
AB	-0.049	0.430
C	0.051	0.408
D	-0.273	0.000
E	0.153	0.015
σ	1.590	0.000

6. ANALYSIS OF THE THERMOSTAT EXPERIMENT

Consider the thermostat experiment in section 2 which studied 11 factors (A - K). Taking the ML approach, a lognormal regression model with 11 factor main effects (A - K) was fit using the lifetime data (table 3); the results are in table 8 (model 1). (At most 12 effects can be fit simultaneously because of the design run size of 12, so that besides an intercept only the main effects could be considered in an initial analysis.) Based on table 8, 9 of the factors appear to be important. One reason for so many apparently important factors was pointed out by Bullington *et al* [3]: Each group of 10 units was produced at the same time so that the variability among the 10 units tends to be smaller than if they had been produced at different times. This reduced variability, which is used in the statistical testing, is one possible explanation for many *s*-significant effects. Some additional analysis based on [8, 9] which account for the properties of the 12-run Plackett-Burman design used in this experiment, suggests the presence of an $E \times H$ interaction. Further evidence of an interaction is seen in table 8 (Model 1) by noting that the MLE for all factors except E & H have nearly the same magnitude. Consequently, model 2 was fit in which the factor B main effect (the least important from model 1) was dropped and replaced by the $E \times H$ interaction. The results in table 8 (model 2) indicate that only the E, H, $E \times H$ effects are important. The original analysis [3] found only E & H important and recommended E_1H_1. The same recommendation is obtained using the table 8 results. While the original analysis did not account for the possibility of an interaction, the same recommendations result because the two factors have a synergistic relationship, *viz*, both increase reliability as they are changed from their level 2 to their level 1. In summary, the analysis suggests that reliability gains can be made by using a grain size (factor E) of 0.008 inches with a heat treatment (factor H) of 45 minutes.

7. ANALYSIS OF THE GOGGLE EXPERIMENT

In the robust reliability experiment to improve night-vision goggles in section 3, there were 10 control-factors (A - J) and 1 noise-factor (N). Taking the response-model approach, a lognormal regression model (1) can be fit which consists of an intercept (INT), 10 Control main effects (A - K), 1 Noise main effect (N), and 10 Control×Noise interactions ($A \times N$ - $K \times N$), where Control & Noise denote control & noise factors, respectively. Using the product-array data in table 6, the MLE for this model do not exist. Consequently, the Bayes approach [10] was taken. Using a relatively diffuse prior, table 9 gives the central 0.95 & 0.99 intervals of the marginal posteriors for each effect. The important effects appearing in bold face are those whose central 0.95 intervals do not contain zero; in fact, the 0.99 intervals for all of these except the $I \times N$ interaction do not contain zero. Based on these results, main effects, B, D - H, and interactions, $A \times N$, $C \times N$, $E \times N$, $I \times N$, $J \times N$, are important.

TABLE 8
MLE and *P* Values of Lognormal Regression Models
[Thermostat Experiment]

Model 1			Model 2		
Effect	MLE	*P* Value	Effect	MLE	*P* Value
A	-0.312	.0001	*A*	-0.091	.3890
B	0.221	.0024	*EH*	0.663	.0024
C	-0.319	.0001	*C*	-0.098	.3474
D	0.285	.0001	*D*	0.064	.5174
E	-1.023	.0001	*E*	-1.023	.0001
F	0.231	.0016	*F*	0.010	.9219
G	-0.390	.0001	*G*	-0.169	.1075
H	-0.557	.0001	*H*	-0.557	.0001
I	-0.332	.0001	*I*	-0.112	.2872
J	-0.277	.0001	*J*	-0.056	.5958
K	-0.352	.0001	*K*	-0.131	.2149

For this experiment, the relationship between the response and the control & noise factors is too complicated to make control factor-level recommendations simply by inspecting the model. Consequently, the criteria in section 3 based on the reliability (3) distribution (mean, standard deviation, and minimum) can be evaluated over the 2 levels of the 1 noise-factor for each of the possible combinations of control factors ($1024 = 2^{10}$) and then ranked appropriately (out of 1024, with 1 being the best). At each control factor setting, the reliability (3) was evaluated using a time $T = 100$ with (β, σ) being sampled 5000 times from the posterior distribution at each noise factor-level which yielded a total of 10^4 reliabilities. Thus, the uncertainty in (β, σ) as described by the posterior distribution is easily incorporated by the Bayes approach, which is an attractive advantage. Table 10 presents the 15 best control-factor combinations according to the mean criterion, and the standard-deviation & minimum criteria. Based on these results, a good choice of factor levels is:

$A_1B_2C_1D_1E_2F_2G_1H_2I_2J_2$ (row #1 of table 10).

Its minimum criterion is the best and its standard-deviation criterion is also small which indicate that these levels are robust to the noise factor. Contrast this with the settings in row #7

of table 10 which has a relatively high mean but is clearly not robust to the noise factor.

TABLE 9
Posterior Quantiles Using Diffuse Prior
[Goggle Experiment]

	Quantile			
Effect	.005	.025	.975	.995
INT	2.60	2.64	2.99	3.07
A	-0.15	-0.08	0.32	0.40
B	0.74	0.80	1.15	1.24
C	-0.23	-0.18	0.23	0.28
D	-0.79	-0.71	-0.32	-0.24
E	0.31	0.37	0.77	0.84
F	0.04	0.10	0.52	0.60
G	-0.60	-0.52	-0.12	-0.07
H	0.27	0.32	0.70	0.82
I	-0.35	-0.27	0.12	0.20
J	-0.25	-0.18	0.23	0.30
N	-0.11	-0.05	0.29	0.35
AN	-0.68	-0.62	-0.21	-0.16
BN	-0.12	-0.05	0.29	0.35
CN	-0.82	-0.72	-0.34	-0.27
DN	-0.22	-0.17	0.24	0.31
EN	-0.58	-0.50	-0.07	-0.01
FN	-0.09	-0.03	0.39	0.45
GN	-0.46	-0.37	0.03	0.09
HN	-0.35	-0.24	0.16	0.23
IN	-0.53	-0.44	-0.04	0.03
JN	-0.79	-0.71	-0.33	-0.27
σ	0.02	0.02	0.07	0.11

ACKNOWLEDGMENT

This research was supported by General Motors of Canada Limited, the Manufacturing Research Corporation of Ontario, and the Natural Sciences and Engineering Research Council of Canada. I appreciate the helpful suggestions of the referees that led to improvements in the paper's exposition.

REFERENCES

[1] G.E.P. Box, N.R. Draper, *Empirical Model-Building and Response Surfaces*, 1987; John Wiley & Sons.

[2] G.E.P. Box, W.G. Hunter, J.S. Hunter, *Statistics for Experimenters*, 1978; John Wiley & Sons.

[3] R.G. Bullington, S.G. Lovin, D.M. Miller, W.H. Woodall, "Improvement of an industrial thermostat using designed experiments", *J. Quality Technology*, vol 25, 1993 Oct, pp 262-270.

[4] L.W. Condra, *Reliability Improvement with Design of Experiments*, 1993; Marcel Dekker.

[5] A. Dey, *Orthogonal Fractional Factorial Designs*, 1985; Halsted Press.

[6] M. Hamada, "Reliability improvement via Taguchi's robust design", *Quality & Reliability Engineering Int'l*, vol 9, 1993 Jan/Feb, pp 7-13.

[7] M. Hamada, S.K. Tse, "On estimability problems in industrial experiments with censored data", *Statistica Sinica*, vol 2, 1992 Jul, pp 381-391.

[8] M. Hamada, C.F.J. Wu, "Analysis of censored data from highly fractionated experiments", *Technometrics*, vol 33, 1991 Feb, pp 25-38.

[9] M. Hamada, C.F.J. Wu, "Analysis of designed experiments with complex aliasing", *J. Quality Technology*, vol 24, 1992 Jul, 130-137.

[10] M. Hamada, C.F.J. Wu, "Analysis of censored data from fractionated experiments: a Bayesian approach," *J. Amer. Statistical Assoc*, vol 90, 1995 Jun, (to appear).

[11] R.N. Kackar, "Off-line quality control, parameter design, and the Taguchi method (with discussion)", *J. Quality Technology*, vol 17, 1985 Oct, pp 176-209.

TABLE 10
Best Factor Settings
[Goggle Experiment]

Settings										Criterion					
										Mean		Std Dev		Min Prob	
A	*B*	*C*	*D*	*E*	*F*	*G*	*H*	*I*	*J*	Value	Rank	Value	Rank	Value	Rank
1	2	1	1	2	2	1	2	2	2	0.999	1	0.007	224	0.755	1
1	2	2	1	2	2	1	2	2	1	0.999	2	0.011	266	0.521	5
1	2	2	1	2	2	1	2	1	1	0.999	3	0.010	258	0.565	4
1	2	1	1	2	2	1	2	1	2	0.999	4	0.010	255	0.687	2
2	2	1	1	2	2	1	2	1	2	0.998	5	0.014	284	0.487	6
2	2	2	1	2	2	1	2	1	1	0.998	6	0.015	295	0.594	3
2	2	1	1	2	2	1	2	2	1	0.990	7	0.053	398	0.041	15
2	2	2	1	2	2	1	2	2	1	0.987	8	0.060	416	0.127	10
2	2	1	1	2	2	1	2	2	2	0.987	9	0.063	429	0.180	7
1	2	2	1	2	1	1	2	2	1	0.980	10	0.064	430	0.164	9
1	2	1	1	2	2	2	2	2	2	0.980	11	0.066	436	0.070	12
1	2	1	1	2	1	1	2	2	2	0.979	12	0.068	441	0.092	11
1	2	2	1	2	2	2	2	2	1	0.978	13	0.071	447	0.166	8
1	2	2	1	2	2	1	2	1	2	0.971	14	0.099	487	0.020	17
2	2	1	1	2	1	1	2	1	2	0.964	15	0.113	509	0.026	16

[12] J.F. Lawless, *Statistical Models and Methods for Lifetime Data*, 1982; John Wiley & Sons.
[13] D.C. Montgomery, *Design and Analysis of Experiments* (3rd ed), 1991; John Wiley & Sons.
[14] F. Montmarquet, "Printed circuit drill bit design optimization using Taguchi's methods % .013-inch diameter bits," *Sixth Symp. Taguchi Methods*, 1988, pp 70-77; Amer. Supplier Inst.
[15] W. Nelson, *Applied Life Data Analysis*, 1982; John Wiley & Sons.
[16] M.S. Phadke, "Design optimization case studies", *AT&T Technical J.*, vol 65, 1986 Mar/Apr, pp 51-68.
[17] R.L. Plackett, J.P. Burman, "The design of optimum multifactorial experiments", *Biometrika*, vol 33, 1946 Jun, pp 305-325.
[18] H. Raiffa, R. Schlaifer, *Introduction to Statistical Decision Theory*, 1961; McGraw-Hill.
[19] T.L. Reed, "Reduction of gain change and improved humidity reliability in gen III night vision goggle tube sub-assembly using an L_{12} Taguchi orthogonal array", *Sixth Symp. Taguchi Methods*, 1988, pp 247-268; Amer. Supplier Inst.
[20] *SAS/STAT User's Guide*, version 6 (4th ed), vol 2, 1989; SAS Institute Inc.
[21] A.C. Shoemaker, K.L. Tsui, C.F.J. Wu, "Economical experimentation methods for robust design", *Technometrics*, vol 33, 1991 Nov, pp 415-427.
[22] M.J. Silvapulle, J. Burridge, "Existence of maximum likelihood estimates in regression models for grouped and ungrouped data", *J. Royal Statistical Soc*, ser B, vol 48, 1986, pp 100-106.
[23] N. Specht, "Heat exchanger product design via Taguchi methods", *Third Symp. Taguchi Methods*, 1985, pp 302-318; Amer. Supplier Inst.
[24] D. Steinberg, P. Colla, *SURVIVAL: A Supplementary Module for SYSTAT*, 1988; SYSTAT Inc.
[25] G. Taguchi, *Introduction to Quality Engineering*, 1986; Asian Productivity Org.
[26] G. Taguchi, *System of Experimental Design*, 1987; Unipub/Kraus Int'l Publ.
[27] G. Taguchi, Y.I. Wu, *Introduction to Off-Line Quality Control*, 1980; Japan QC Assoc.
[28] J.C. Wang, C.F.J. Wu, "An approach to the construction of asymmetrical orthogonal arrays", *J. Amer. Statistical Assoc*, vol 86, 1991 Jun, pp 450-456.
[29] W.J. Welch, T.K. Yu, S.M. Kang, J. Sacks, "Computer experiments for quality control by parameter design", *J. Quality Technology*, vol 22, 1990 Jan, pp 15-22.

AUTHOR

Dr. Michael S. Hamada; Dept. of Statistics and Actuarial Science; Univ. of Waterloo; Waterloo, Ontario N2L 3G1 CANADA.

Michael S. Hamada is Research Associate Professor in the Department of Statistics and Actuarial Science and for The Institute for Improvement in Quality Productivity at the University of Waterloo. His research interests include design of experiments, reliability, and statistical process control. He received a PhD (1987) in Statistics from the University of Wisconsin - Madison.

Manuscript received 1994 August 30.

IEEE Log Number 94-11783 ◄TR►

DOE/Opt: A System for Design of Experiments, Response Surface Modeling, and Optimization Using Process and Device Simulation

Duane S. Boning, *Member, IEEE*, and P. K. Mozumder, *Member, IEEE*

Abstract—**Rapid modeling and optimization of manufacturing processes, devices, and circuits are required to support modern integrated circuit technology development and yield improvement. We have prototyped and applied an integrated system, called DOE/Opt, for performing Design of Experiments (DOE), Response Surface Modeling (RSM), and Optimization (Opt). The system to be modeled and optimized can be either physical or simulation based. Within the DOE/Opt system, coupling to external simulation or experimental tools is achieved via an embedded extension language based on Tcl. The external problem then appears to DOE/Opt as a model with user defined inputs and outputs. DOE/Opt is used to generate splits for experiments, to dynamically build and evaluate regression models from experimental runs, and to perform nonlinear constrained optimizations using either regression models or embedded executions. The intermediate regression modeling can appreciably accelerate the optimization task when simulation or physical experiments are expensive. The primary application of DOE/Opt has been to process optimization using coupled process and device simulation. DOE/Opt has also been applied to process and device simulator tuning, and to aid in device characterization. Such a DOE/Opt system is expected to augment the use of TCAD tools and to utilize data collected by CIM systems in support of process synthesis. We have demonstrated the application of the system to process parameter determination, simulator tuning, process control modeling, and statistical process optimization. We are extending the system to more fully support emerging device design and process synthesis methodologies.**

I. Introduction

SEMICONDUCTOR process and device design can substantially benefit from the use of simulation and modeling to reduce the cost and time required to develop new or extend existing technology [1]. Technology development, however, requires substantially more than a fundamental simulation capability: tools and methods to assist in exploration of design trade-offs and to optimize a design are becoming increasingly important.

Work on frameworks for the integration of technology CAD (TCAD) tools has focused primarily on underlying representations for the wafer and process, and on the interfaces between simulation tools and these representations [2]–[4]. Systems which achieve integration and help automate the execution of tools have also been demonstrated [5]–[7]. Less common are tools or systems which directly address how one uses such integrated simulation capability to solve design problems [8]. Alvarez explored statistical modeling methods for the design and optimization of processes and devices [9]. In that work, formal (Box–Behnken) design of experiments were used to construct response surface models. A grid search method was used to explore the design space and to identify feasible regions given goals and constraints on multiple performances and performance sensitivities. Other design exploration and optimization techniques have been investigated elsewhere, particularly for circuit yield optimization [10]–[12].

This paper makes contributions in two areas. First, we describe approaches for the use of experimental design, regression or direct simulation modeling, and optimization that we have found to be important in process/device design, simulator tuning, coupling to process control, and design for manufacturability. Second, we describe a solution to issues in the implementation of a general purpose optimization tool suitable for end TCAD users including graphical interface, user programmability via an extension language, and interfaces to existing simulation tools. DOE/Opt is a "task level" tool which complements existing TCAD Framework research on "data level" representations, and builds on "tool level" simulation capability [13]. Two audiences should thus benefit from this paper: 1) designers who will increasingly demand the types of simulation, modeling, and optimization capability represented by DOE/Opt, and 2) implementors who will find it necessary to implement similar high level utility programs in the future.

In Section II, the overall architecture of the prototype DOE/Opt system is introduced. Each of the key conceptual components of the system is discussed in succeeding sections. The encapsulation and use of simulation or other models, and the construction of analytic response surfaces is discussed in Section III. The integration of formal experimental design techniques with 1) simulation or experimental execution, and 2) design exploration and model construction are described in Section IV. The integration and use of nonlinear optimization is discussed in Section V. Four examples are presented in Section VI to illustrate the use of the DOE/Opt system. These include process/device performance optimization, simulator model tuning, process control recipe generation, and manufacturability optimization. Conclusions are drawn in Section VII.

Manuscript received August 1, 1993; revised December 7, 1993. This work was supported in part by the Air Force Wright Laboratory and the DARPA Microelectronics Technology Office under Contract F33615-88-C-5448.

D. S. Boning is with the Department of Electrical Engineering and Computer Science, Massachusetts Institute of Technology, Cambridge, MA 02139 USA.

P. K. Mozumber is with the Semiconductor Process and Design Center, Texas Instruments, Inc., P.O. Box 655012, Dallas, TX 75235 USA.

IEEE Log Number 9216485.

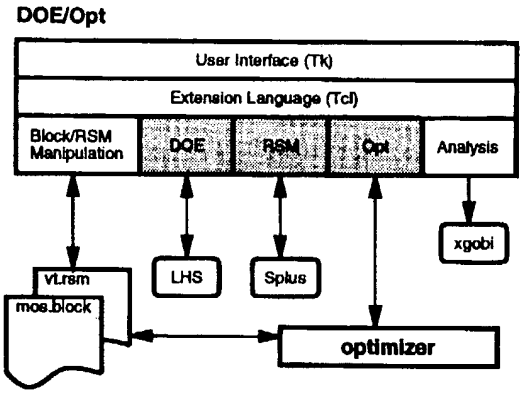

Fig. 1. The architecture of the DOE/ Opt system.

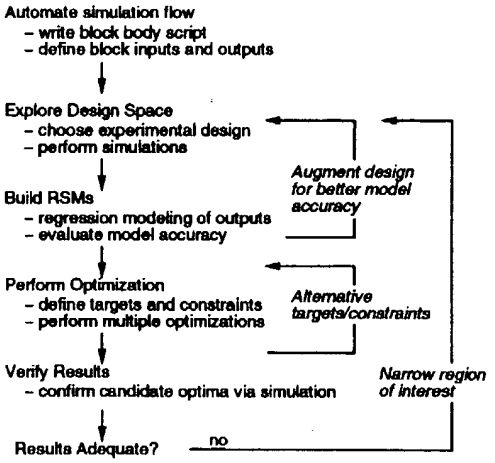

Fig. 2. Flow diagram of typical DOE/Opt system use.

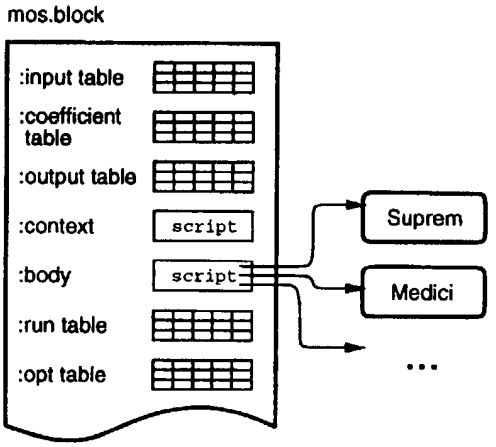

Fig. 3. The conceptual organization of a DOE/Opt block.

II. System Architecture

The DOE/Opt system architecture is summarized in Fig. 1. There are three key computational components to the system: 1) design of experiments (DOE); 2) response surface model generation (RSM); and 3) nonlinear constrained optimization (Opt). The DOE/Opt executable uses the Tcl [14] extension language for application and end-user programmability. On top of this sits a Motif compatible graphical user interface (GUI) implemented in Tk [15] to manipulate models. Through the GUI, the end-user directs the execution of model blocks, constructs response surface models, and defines and directs optimizations. Optimizations are performed by a second executable program which implements multiple objective nonlinear constrained optimization on top of the NPSOL [16] package. DOE/Opt consists of approximately 10 000 lines of Tcl/Tk code, and about 1 000 lines of C code (to interface to NPSOL).

A typical DOE/Opt work flow used to explore or optimize design trade-offs is shown in Fig. 2. The first task is to automate, via a block script, the simulation and data extraction flow. Second, one chooses an appropriate experimental design to systematically vary block inputs, and performs the corresponding simulations. Third, one constructs and evaluates response surface models for the outputs. Additional design points may need to be simulated and different modeling strategies employed to achieve an adequate model fit. Once achieved, target values and constraints on the outputs are defined, and optimization using the response surface models (or using the body directly) are performed. One or more candidate optima are verified using the full simulation, which may indicate the need for additional simulation and model building in the region of interest.

In the following sections, the key components in the DOE/Opt architecture will be described in more detail. In Section VI, variations on the usage flow of Fig. 2 will be discussed for a number of optimization scenarios.

III. Model Representation

Experimental design and optimization fundamentally require a way to express the response of the system under study. Conceptually, a generic representation of a *model* of that system is needed, independent of whether analytic, numeric, or experimental execution of the model produces the response. In this way, many of the same experimental design, execution, and optimization mechanisms can be applied to real systems, to extensive numerical simulations, or to fast analytic models.

In DOE/Opt, a *block* is the embodiment of some model, and maps inputs to outputs. A block has several tabular and script components as pictured in Fig. 3. The key computational component is the block *body*. The block body is often a script that chains together process simulation, device simulation, and output data extraction. In other cases, a response surface model (or *RSM block*) contains a block body which is an analytic polynomial function to calculate response and gradient information. Similar to the block body is the block *context*, which is another script that is run only once when the block is first loaded. The context is useful for defining utility procedures that will be called repetitively within the block body.

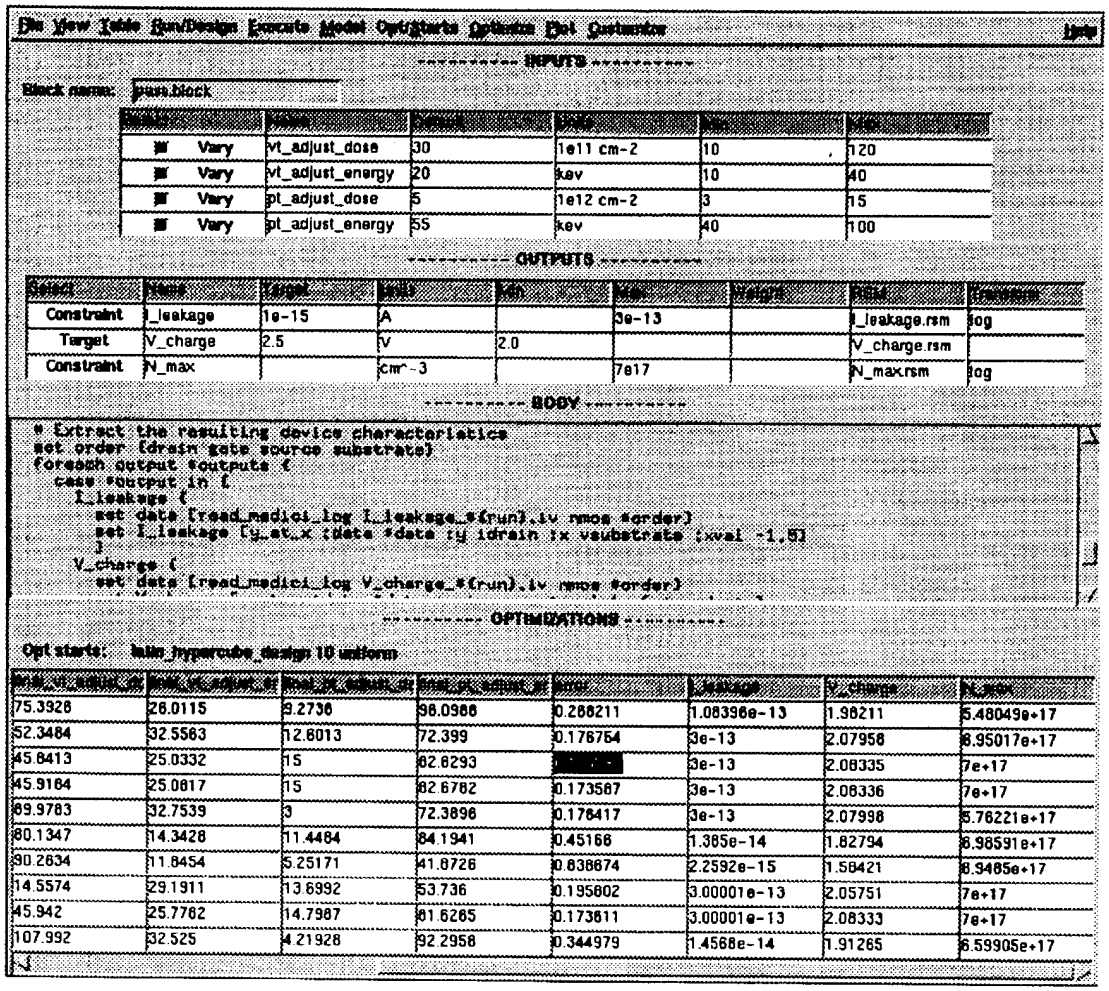

Fig. 4. The DOE/Opt user interface. Pictured is a block for the calculation and optimization of key 256 Mb DRAM pass transistor parameters, including display of inputs, outputs (targets and constraints), a portion of the simulation script in the block body, and optimization results.

The block *input table* defines the possible inputs to the block. For each input, the name, default value, units, minimum, and maximum values may be specified. A toggle for each input indicates whether or not that input will vary in an experimental design or optimization. The block *coefficient table* is very much like the input table, except that it defines the values of any internal model coefficients inside the block. For example, automatically generated response surface models have coefficient values determined by the regression fit to run data. The block *output table* defines the possible outputs that the block is able to compute. Each output has a name, and may have an associated "RSM" giving the filename of an external response surface model that may be used in place of the block body to calculate the output. Desired transformations of output values may also be specified. For optimization, additional information about each output may be specified in the output table: each output may have a target value, and/or may have upper and lower constraint values. Finally, each row in the run and optimization tables presents specific input values and resulting outputs (or the initial starting point and resulting optima in the case of the optimization table).

A. Concrete Representation

We represent each block as a separate UNIX file. The block file contains information that may be loaded into the DOE/Opt user interface where it can be viewed, edited, executed, and saved back to a UNIX file. The DOE/Opt graphical user interface for a typical block is shown in Fig. 4. This block file is also directly executable and manipulatable from the UNIX command line; this is crucial for the use and manipulation of blocks in other scripts, programs or systems external to DOE/Opt. For example, a block which chains process and device simulation to calculate threshold voltage might be executed from the UNIX command line, where input values are specified as command line arguments and the result is

returned by the block:

```
unix> vt.block -ox_time 100
                    -channel_length 0.5
0.675
```

Alternatively (and more commonly), the same execution might be performed directly from the user interface. In this case, one would define an experiment via the "Run/Design" pulldown menu which would generate new rows in the run table. The user might optionally override the inputs, coefficient, or run table entries for the `ox_time` and `channel_length` variables, and then execute the run table entries via the "Execute" pulldown menu. In this case, the block body would be run and the results displayed in the output columns of the run table.

The block may be manipulated in other ways, and an entire set of runs and optimizations (as defined per some experimental design) can be invoked, either from the graphical user interface or from the UNIX command line. For example, to change the target value for the threshold voltage output, and then to perform the optimization specified inside the block, one could use the unix command line commands:

```
unix> vt.block set output vt -target 0.7
unix> vt.block optimize_rsm.
```

In the graphical user interface, one would simply point to the entry widget in the `target` column for the `vt` output row to change the value, and then initiate the optimization via the "Optimize using RSM's" option in the "Optimize" pulldown menu. This performs the optimization using response surface models for the desired outputs (`vt` in this case), and fills in the optimization table with the optimal operating points. In the UNIX invocation, the resulting optimization table will reside within the block file at the completion of the optimization. To support long simulation or optimization problems, it is critical that the graphical user interface not be required; a UNIX command line approach, for example, enables overnight runs without a graphical display.

B. Encapsulation of Simulation Flows

We have made two specific uses of the generic block interface. One is to represent automatically constructed response surface models as described later in Section IV-C. The other is to automate simulation and data extraction [17]. To support the user definition of simulation flows, the following four issues are important. First, a high-level interpreted scripting language is essential for dynamic generation and use of scripts. We use the Tcl (Tool Command Language [14]) extension language for all block body scripts, thus ensuring that full language mechanisms (e.g., procedure definitions, variable manipulations, mathematical expression evaluation, and control flow) are available as needed. Second, the script must be well connected to the surrounding block, as that is where inputs, coefficients, and outputs are specified. We accomplish this by making all input and coefficient values specified via the GUI (as shown in Fig. 4) available as Tcl variables in the block body script, and by returning requested outputs back from the script to the GUI. Third, "template" versions of simulation input files are important. The parameterization of simulation decks, along with parameter substitution to produce actual simulation input files, is common practice [9]. We extend this so that full expression evaluation, and indeed Tcl procedure invocation, is also possible within the template. For example, a SUPREM3 [18] deck might be parameterized as

```
Initialize  silicon (100) boron=1e15
             Thickness=2
Diffusion   time=#[$ox_time*60.0]#
             temp=1000 dryO2
Implant     arsenic dose=1e15 energy=$=
            {energy}
```

so as to convert time in hours to time in minutes for use in the simulation. An interactive utility program to convert conventional input files to DOE/Opt templates has also been implemented, where selected ranges of text are turned into input variables and the appropriate input item is dynamically added to the block definition in the DOE/Opt GUI. Finally, we have found that interactive control over block execution is important. A "monitor" pop-up can be made to appear for selected simulations within a block body or for the block body execution as a whole. The monitor shows the current execution status and permits termination of problem executions or optimizations.

IV. Design of Experiments

The value of statistically based experimental designs (the matrix of runs generated by specific combinations of input values) has been well established [9], [19]. The DOE/Opt system strongly encourages the use of designed simulation experiments where the input parameters are varied in a carefully structured pattern so as to maximize the information that can be extracted from the resulting simulations. In our work, we have achieved tight integration between specification of models (via blocks), generation of experimental designs, automatic execution of those designs, and generation of response surface models from execution results.

A. Experimental Design Suite

A variety of experimental designs are incorporated into DOE/Opt [19]. A *nominal design* generates a single run with nominal (default) values for all selected block inputs. A *center point design* is similar, but uses the midpoint value (between the maximum and minimum limits) for all selected inputs. Note that the range for input values may be specified directly, as deltas from the default value (e.g., "+5"), or as percentages from the default value (e.g., "−10%"). The *axial design* includes center points, and points at the minimum and maximum of each input while holding other inputs to their nominal values. The *Box–Wilson* design includes the center point, and axial points and corner points distributed around an n-dimensional sphere circumscribing the cuboid defined by the input minima and maxima [20]. We have found two variants

of the conventional *Box–Wilson* design to be useful. The *Box–Wilson (inscribed)* modifies the design by distributing axial and corner points on an n-dimensional sphere inscribed within the cuboid [21]. This has the benefit that no experimental points will be outside the input ranges, but results in models that are not well defined in the corners of cuboid. The *Box–Wilson on a cube* stretches the corner points to lie on the cuboid corners to achieve more complete coverage of the input parameter space. The *full-factorial design* produces a uniform grid with user-specified density covering the input parameter space. Other conventional designs, including *Box–Behnken* and *half fractional*, (as well as the half fraction complement for augmenting a screening design), are provided [19]. Finally, *Latin hypercube sampling* (LHS) [22] provides an orthogonal array that randomly samples the entire design space broken down into r^n equal-probability regions (where r is the number of runs, and n is the number of input variables). LHS can be looked upon as a stratified Monte Carlo sampling where the pairwise correlations can be minimized to a small value (which is essential for uncorrelated parameter estimates) or else set to a desired value [23]. LHS is especially useful in exploring the interior of the parameter space, and for limiting the experiment to a fixed (user specified) number of runs. All designs are generated algorithmically, with the exception of Box–Behnken designs which use table lookups. Latin hypercube samples are generated via an interface to LHS software from Sandia National Laboratories [24].

B. Integration with Block Execution

When an experimental design is selected, the DOE/Opt system examines the block to determine which inputs have been selected to vary, and to determine the input defaults and limits. Based on the design a *run table* is generated (or added to) with empty slots for each of the outputs to be calculated. The run table can then be executed in total or incrementally to fill in missing elements. The run table may be edited after its creation and before executing the block, so that the user has control over the simulation run value, and can delete or add new run table rows to eliminate or perform other simulations. After execution, the run table can be examined using interactive data visualization methods. In the current implementation we interface to the *xgobi* package, an interactive dynamic graphics program which is effective in the exploration and visualization of multivariate data [25]. Finally, response surface models can be automatically generated from the run table.

C. Response Surface Models

An important part of the DOE/Opt system is the generation and use of response surface models to inexpensively "mimic" the more complex workings of a simulation or experimental block. The following four elements of response surface model construction have been found to be important. First, a variety of polynomial base models (linear, quadratic with and without cross terms, cubic, etc.) must be provided. We have found it useful to link the regression model type with the experimental design used (e.g., one builds a "Box–Wilson" response surface model from run data generated via the Box–Wilson design). In addition, scaling of the inputs to the range $[-1, 1]$ is used to increase model construction robustness. The covariance of the estimates, a metric of model stability, is dependent on the input design matrix and the lack of model fit. Choice of the input design matrix is critical to determining the model coefficients, and minimizing the covariance between the estimates of the model coefficients. Scaling the inputs minimizes the correlation between the estimates of the coefficients of the model [26], [27]. Additionally, transformation of outputs may be used (e.g., *log*, *exp*, *square*, *square root*, *inverse*) to aid model fitting and later optimization. Second, response surface models must be capable of generating both response and response gradient information in order to reduce evaluations required during optimization. The automatically generated response surface models have as their output the basic response, followed by the Jacobian with respect to the input parameters, followed by the Jacobian with respect to the model coefficients (to aid in model tuning if desired). Code to calculate all of these analytically is part of the generated response surface model. Third, weighted regression is important: a common sequence is to perform a broad experimental design, build a model, optimize to some region within that model, then refine the experimental design near that region. In such cases, it is useful to weight the second set of simulations more than the first to increase model accuracy in the region where the model will be most used. The response surface models are generated using weighted least squares regression, currently via an interface to the Splus statistical package [28], [29]. Finally, tight integration between the generated model and DOE/Opt is necessary. This includes the ability to perform executions and optimizations using either a full simulation or using generated response surface model blocks. It is also important that response surface models be generated on the fly: a compile, link, and re-execution of the application is unacceptable. For this reason, we again elected to generate interpretive, Tcl-based regression models.

V. Optimization

A key role of the DOE/Opt system is to provide an easy to use optimization capability. Real design and manufacturability optimization problems often have the following features. First, the optimization problems are often nonlinear, and multiple targets or objectives must be reconciled simultaneously. Second, numerous constraints apply in the optimization problem; these constraints may also be nonlinear in nature. Many design and optimization problems focus on choices of continuous parameter values. While discrete value or integer optimization is also often desirable (e.g., to make a choice in whether to include a design feature or not, or to decide between two alternative treatments), we have focused entirely on the continuous parameter optimization problem.

We have constructed a layered optimization capability as shown in Fig. 5. The fundamental optimization capability is provided by the NPSOL package [16]. On top of this a set of Tcl bindings provide interpretive access to the FORTRAN NPSOL routines. In addition to enabling one to dynamically "call into" the optimizer, the optimizer also "calls out" to the Tcl language layer for the evaluation of objective and

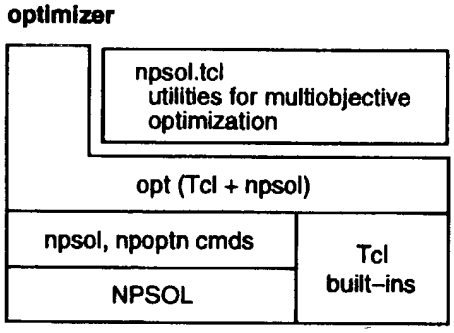

Fig. 5. Layering of the Tcl/NPSOL interface. The higher components make use of the lower level blocks. The shaded components are described in this document.

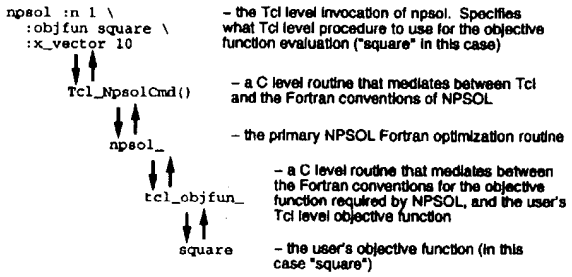

Fig. 6. Detail of the calling sequence in the Tcl to NPSOL interface.

constraint functions, as illustrated in Fig. 6. In this way it becomes possible to define an objective procedure and call the optimizer (which in turn evaluates the objective function several times), all dynamically and at run time. Layered on top of the basic access to NPSOL via Tcl, a generic weighted multiple target, multiple constraint optimization capability has been implemented. This layer is responsible for folding together and making use of all existing model and model gradient information (such as that provided by the DOE/Opt generated response surface models) to reduce optimization solution times. Through this interface, DOE/Opt is able to generate models and optimization problems, and solve them via NPSOL.

Several key issues have been identified and addressed in order to make the tool more appropriate for use in semiconductor process and device design. First, in practice optimization problems can become highly nonlinear and complex. No guarantees on global optimality are made. Instead, an approach to easily support the optimization from numerous starting points is used. In this case, we use the same design of experiments methods outlined in Section IV to produce well distributed coverage of the parameter space in searching for potential optima. The Latin hypercube sample has proven particularly useful in generating optimization starting points.

Second, we have found that a selection of overall objective functions needs to be provided. The current system includes the following choices: weighted sum of squares, weighted sum of normalized squares, weighted sum of absolute errors, weighted cumulative errors, and weighted normalized cumulative error. While some of these are not necessarily well-behaved (e.g., sum of absolute errors), they often map well onto the conceptual optimization problem at hand and work satisfactorily.

Third, we have found that it usually makes sense for the optimization to use transformed response surface model information rather than the "pure" values returned by models constructed under transformations. For example, a *log* transform selected when building a response surface model might fit 10^{14}, 10^{15}, and 10^{16} as a straight line internally. However, the model will still return exponentially large values when evaluated. For optimization, on the other hand, the linear underlying model is more effectively used. The ability to perform model regressions and optimizations in transformed space is important.

VI. Examples

Four examples demonstrate the application of the DOE/Opt system. These include process/device design, simulator tuning, process control recipe generation, and statistical process/device design.

A. Process/Device Optimization

A very common problem in coupled process and device design is to choose process parameters so as to achieve desired device performance. DOE/Opt has been applied to study the selection of channel doping in the design of a 256Mb DRAM pass transistor. The choice of energies and doses for threshold adjust and punchthrough implants (*vt_dose, vt_energy, pt_dose, pt_energy*) are the input parameters under study. The performance parameters are leakage current (*I_leakage*), peak channel doping (*N_max*), and charge voltage (*V_charge*, or V_D at which $I_D > I\_critical$, with $V_G = 3.5$ V, $V_S = 2.5$ V, and $V_B = -1.5$ V), which is a measure of the rate at which the storage capacitor can be charged.

A simulation script executes the SUPREM3 process simulation and MEDICI [30] device simulation sequence given a set of process parameter values; part of this script is shown in Fig. 4. A Box–Wilson on a cube experimental design was first run over a relatively broad input parameter range, and quadratic models constructed. The *V_charge* and *N_max* models fit the data well. The good *N_max* fit is expected, since the maximum doping concentration in a single implant is directly proportional to implant dose and inversely proportional to implant straggle (which is only a weak function of implant energy). The *V_charge* fit is also not surprising, as there is an inverse dependence of *V_charge* on $V_G - V_T$ for small V_{DS}, and V_T is roughly linearly dependent on implant dose; the resulting inverse dependence is weak enough for our conditions that it can be well approximated by a quadratic response surface model. However, the *I_leakage* model required the use of logarithmic transforms, and still achieved a relatively poor fit due to highly nonlinear behavior when punchthrough occurs. By weighting the simulation runs

prior to regression, an adequate fit near the *I_critical* parameter was achieved.

Based on these models, an optimization was performed with a high *V_charge* output selected as the target (where we minimized the squared error from the target), with constraints on *I_leakage* and *N_max* as indicated in Fig. 4). The resulting trade-off between *I_leakage* and *V_charge* is clearly seen in Fig. 7, which also shows the original design points and candidate optima (possible optimal points suggested by the nonlinear optimization using the response surface models) for the threshold adjust parameters. Additional iterations of simulation, model building, and optimization as in Fig. 2 could be used to obtain and verify optima to a desired level of accuracy. An interesting conclusion based on study of the two implants is that both implants share functionality in setting the threshold and protecting against punchthrough. Future work using DOE/Opt will study the necessity of two channel implants as compared to a single channel adjust implant. We also note that the feasible region appears to be narrow; the need for manufacturability analysis as described in Section VI-D is clearly indicated.

B. Simulator Tuning

DOE/Opt has been applied to simulator model tuning. In this case, the goal was to determine ionization coefficients in an internally enhanced version of PISCES [31] to match measurements of substrate current. The model is of the form $\alpha = \alpha^{\infty} \exp\left[-\left(E_{\text{crit}}/E_{||}\right)^{n}\right]$. PISCES simulation decks were developed which calculate substrate current (I_S) at drain voltages $V_D = 5, 6$, and 7 volts, and gate voltage $V_G = 1, 2, 3$, and 4 volts. The input parameters to the script are the ionization model coefficients α^{∞}, E_{crit}, and exponent n. A Box–Wilson experimental design varying these parameters was performed, and regression models of $I_S(V_G, V_D)$ as a function of the ionization model coefficients were constructed. Optimization was performed using the response surface models to find ionization model coefficients that achieved the best fit to the experimental data (minimum sum of squared error). The result was approximately 15% average error across several V_D values. A comparison between the substrate currents from measurements, simulation before calibration, and simulation after calibration is made in Fig. 8.

Two observations on the use of DOE/Opt in this problem bear special mention. First, we found that DOE/Opt is well suited to optimization and fitting to a small and mixed number of targets and constraints. In its current form, it is not ideal for "parameter extraction" where one desires to fit to large sets of data. Work to improve DOE/Opt's capability by interfacing to the MACH system is underway [32]. Second, we find that the most difficult task facing a user of DOE/Opt is writing the data extraction routines used within a script body. We found it necessary to develop and make available to users utility routines that extract data (e.g., current-voltage curves) from process and device simulation output files. To support higher level tools such as DOE/Opt, it is important that the textual and binary output formats of simulation tools be well documented and available.

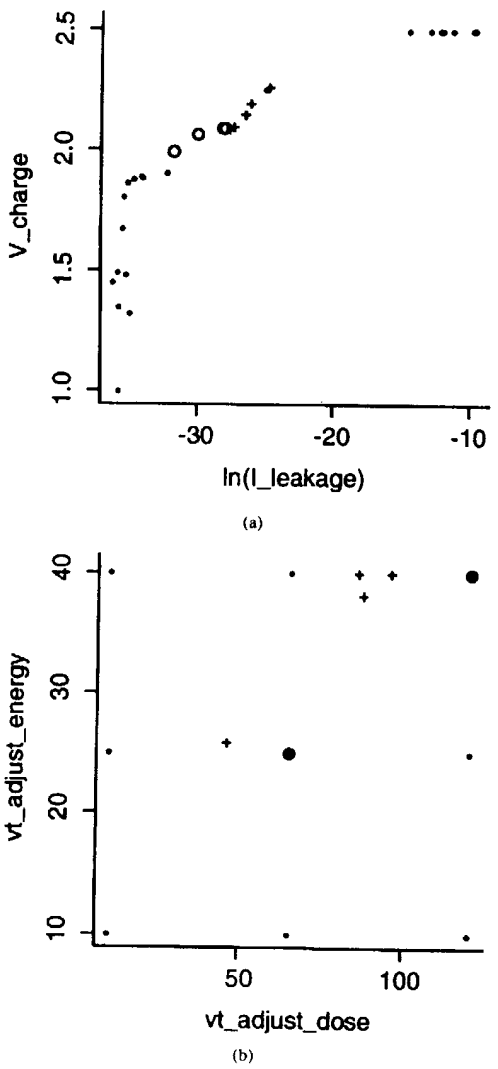

Fig. 7. Exploration of 256-Mb pass transistor channel implant design. The trade-off between leakage and charging voltage is shown in (a). The corresponding threshold adjust design points are shown in (b). The original Box–Wilson experimental design points are shown as circles (feasible points satisfying the design constraints are shown as open circles), and the candidate points from optimization are shown as pluses.

C. Process Control Recipe Generation

DOE/Opt has been used to generate optimal recipes for a plasma enhanced chemical vapor deposition of silicon nitride (PECVD nitride) process run on the Applied Materials Precision 5000 reactor (AMT5000). As a part of the Microelectronics Manufacturing Science and Technology (MMST) program our aim was to model (physically and empirically), optimize, and adaptively control the process to target [33], [34]. We have used DOE/Opt to generate the initial recipe for the process. Using the models of the PECVD nitride process, the optimal equipment settings for the AMT5000 were generated to obtain the desired outputs from the process.

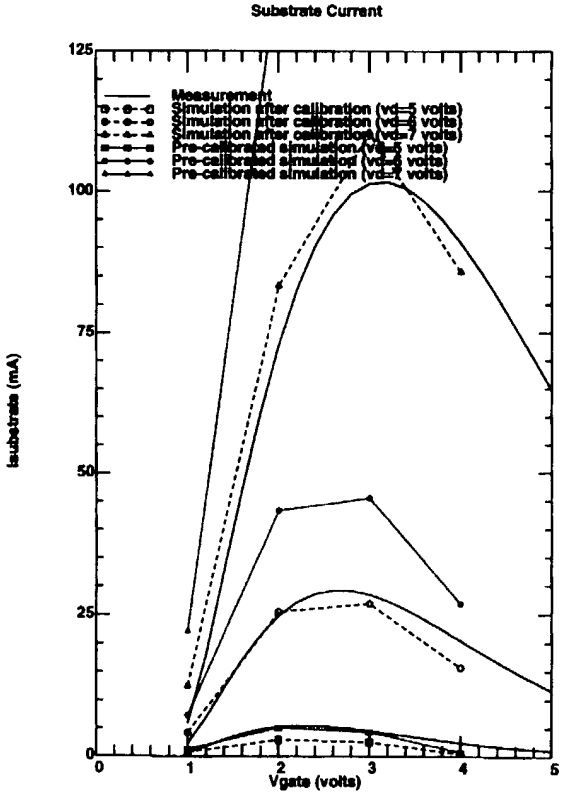

Fig. 8. Comparison of experimenta substrate current, modeled substrate current using calibrated model coefficients, and modeled current using default (pre-calibrated) model coefficients.

The DOE/Opt setup for recipe generation is shown in Fig. 9. The equipment controllables are specified via the input table. The primary equipment controls on the AMT 5000 for the PECVD nitride by the nitrogen–silane–ammonia process are the N_2 flow (N_2), SiH_4 flow (SiH_4), NH_3 flow (NH_3), pressure (Pr), RF power (F), and electrode gap (Gap). The ranges are chosen by the process engineer based on his understanding of the limitations of the equipment and validity of the process models. The default values are the centers of the hyperbox defined by the ranges. The quality characteristics of interest include the film deposition rate ($rate$), index of refraction (ri), stress ($stress$), and thickness nonuniformity (nu). These responses, the corresponding desired values, and the specification limits are indicated in the output table. The block body contains the Tcl script used to encapsulate the pre-existing PECVD nitride model program (invoked using the C executable `pecvd`). The body script handles the conversion to and from that expected by the `pecvd` program.

A Box–Wilson on a cube experimental design was used to create full quadratic response surface models. The *run table* comprised 77 rows, each of which corresponded to a point in the input design space. For each of the rows the model was evaluated and all four output values were obtained by executing the body. Once the run table is filled, regression

Inputs

vary	name	default	units	min	max
1	N2	2000	SCCM	1800	2200
1	SiH4	160	SCCM	140	160
1	NH3	40	SCCM	33	49
1	Pr	4.8	Torr	4.3	5.3
1	RF	370	Watts	317	426
1	Gap	0.42	inches	.405	.435

Outputs

select	name	target	units	min	max	weight	rsm	transform
Target	rate	132		130	134	1.24	rate.rsm	
Target	stress	-120		-170	-70	0.008	stress.rsm	
Target	ri	2.1		2.05	2.15	150.83	ri.rsm	
Constraint	nu			0	2	9.64	nu.rsm	

Body

```
set xin [list $N2 $SiH4 $NH3 $Pr $RF $Gap]
vec_write /tmp/si3n4_[id process] xin
exec csh -c \
 "pecvd < /tmp/si3n4_[id process] >& /tmp/si3n4_[id process].out"
vec_read /tmp/si3n4_[id process].out xout

set xout [unlist $xout]
set rate   [lindex $xout 0]
set stress [lindex $xout 1]
set ri     [lindex $xout 2]
set nu     [lindex $xout 3]
order_results
```

Fig. 9. DOE/Opt setup for recipe generation of the PECVD nitride process on the AMT5000.

is used to generate full quadratic response surface models (consisting of 28 coefficients) for each of the outputs in terms of all six inputs. The generated RSM block files are linked to the optimization problem via the "RSM" column in the output table.

A weighted sum of squares optimization is performed to determine the optimal recipe for the process. The weights for the outputs are chosen as the inverse of the standard errors of the models (see [34] for rationale). For the PECVD nitride process *rate*, *stress*, and *ri* are targets, whereas *nu* acts as a constraint (the specific target and constraint values are shown in Fig. 9). The objective function is specified to be a weighted sum of squares of the difference between the model prediction and the target. In one instance of recipe generation, all of the inputs were allowed to vary for the optimization, optimal points were found and verified, and the controller was initiated with the generated initial recipe; the controller was subsequently able to adjust the recipe to response to equipment shifts and drifts [34]. A second instance of recipe generation is described here, in which the process engineer wanted to generate an optimal recipe where the *Gap* was not changed from its value of 0.42 (either for the initial recipe or during control). Minimal changes had to be made to the DOE/Opt block to generate the initial recipe under the new constraint. The "Vary" button corresponding to the *Gap* in the input table was deselected (to toggle the "vary" value for the gap in Fig. 9). A new set of starting points, without the variable *Gap*, was then generated for the optimizer. To minimize the probability of getting a solution at a local minimum, multiple starting points were used for the optimization: the nominal point and a 10 point Latin hypercube design was used. Corresponding to each of the starting points an optimal point was generated. Two distinct optima were found where the objective function values

—— Verification Runs ——

run	N2	SiH4	NH3	Pr	RF	rate	stress	ri	nu
2	1800	160	49	5.12465	390.149	130.21	-123.379	2.08024	0.703854
4	2200	160	49	4.97737	392.69	130.141	-122.537	2.06386	0.688519

Fig. 10. Verification of optimal recipe for the PECVD nitride process on the AMT5000 with *Gap* held constant.

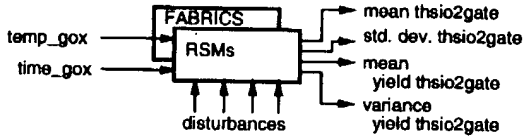

Fig. 11. Schematic of manufacturability simulation.

are relatively small: for $N_2 = 1800$, $SiH_4 = 160$, $NH_3 = 49$ (with error 0.025) and $N_2 = 2200$ (with error 0.028). Both of the optima are verified using the PECVD nitride model, and the results are shown in Fig. 10. Run #2 was chosen as the best recipe since it had a closer to target value of *ri* by trading off a small amount on *stress*. The controller was activated using the initial recipe generated by DOE/Opt. The controller is presently in routine use [34].

D. Design for Manufacturability

DOE/Opt was used with a statistical process simulator to study design for manufacturability. Several criteria for optimizing process recipes under random process variations have been described in the literature [35]–[39]. The concepts of parametric yield, standard deviation and "signal to noise ratio" have been implemented using DOE/Opt and used for process optimization. We present an example of gate oxidation process optimization under simulated manufacturing variations. Using this example, we compare three different design for manufacturability strategies.

The semi-analytic statistical process and device simulator FABRICS [40], [41] is used to emulate the effects of process and material parameter *disturbances* on the resulting device structure. Given the nominal settings for a process step and the standard deviations for the process disturbances, we seek to determine the mean shift in the disturbances (or equivalently new process parameter nominal values) that will result in a design that is optimal under some manufacturability criterion.

We encapsulated FABRICS into DOE/Opt via a body Tcl script which generates FABRICS input files, runs the simulator, and parses the resulting output file. For each DOE/Opt run, FABRICS is executed in the statistical mode to generate 100 samples, from which the statistical response parameters are calculated. As shown in Fig. 11, our approach is to construct response surface models for each of several responses as a function of the inputs and disturbances. In the oxidation example below, a Box–Wilson on a cube design was executed and full quadratic models for each output were generated using least squares regression. We found that all of the models had excellent goodness of fits.

Gate Oxidation Problem Formulation: Fig. 12 shows the DOE/Opt problem formulation for optimizing a gate oxidation process. The set up is analogous to that described in the previous section. The input table consists of six variables. There are two process parameters: time of gate oxidation (*time_gox*), and temperature of gate oxidation (*temp_gox*). Associated with each of the process parameters are the mean and the standard deviation for the disturbances, denoted by the prefixes *md_* and *sd_*. The means and the standard deviations for the disturbances, which are assumed to be independent and normally distributed, can be determined by tuning FABRICS to the fabrication line for which the process is to be optimized [42]. In using DOE/Opt, we have explicitly specified the means and standard deviations of the disturbances corresponding to *time_gox* and *temp_gox*, and do not vary the standard deviations (i.e., they are fixed at their tuned values). The output table consists of five parameters. These correspond to the mean of the gate oxide thickness (*mean_thsio2gate*), standard deviation of the gate oxide thickness (*sig_thsio2gate*), signal-to-noise ratio of the gate oxide thickness (*s2n_thsio2gate*) defined as the ratio of the mean to the standard deviation, mean of Monte Carlo parametric yield for the gate oxide thickness (*ym_thsio2gate*), and the standard deviation of Monte Carlo parametric yield for the gate oxide thickness (*ysd_thsio2gate*). The standard deviation is used to determine the confidence interval on the yield based on the Monte Carlo sampling. Yield calculation is based on whether the resulting responses fall within a region of acceptability; the limits for the acceptability of the gate oxide thickness are specified using the DOE/Opt coefficients table.

Gate Oxidation Optimization Results: Three different objective functions were used as manufacturability criteria for the gate oxidation optimization. The aim was to observe if one particular metric was better than the others for solving the problem of statistical recipe generation. Following are the objective functions used as the test cases:

Case 1: Min *sig_thsio2gate* (std. dev.)
Case 2: Max *s2n_thsio2gate* (signal to noise)
Case 3: Max *ym_thsio2gate* (Monte Carlo yield)
s.t. *ysd_thsio2gate* $\leq 5\%$
All cases: s.t. $4.1 \leq$ *mean_thsio2gate* ≤ 4.3

The minimization and maximization were carried out using a small and a large target value, respectively. For each case, optimizations (smallest sum of square errors from defined targets) were carried out from multiple starting points. The verification runs of the best optima for each case are shown in Fig. 12.

It was found that for each of the three cases the optima for all the starting points resulted in very nearly the same values for the outputs. Moreover, the final values of the output parameters are almost identical irrespective of the objective function. On closer observation it is also found that the optimal process inputs, *md_time_gox* and *md_temp_gox*, are identical for the first two cases, i.e., where *sig_thsio2gate* and *s2n_thsio2gate* are used as objective functions. The optimal process inputs when yield was used as an objective function are slightly different.

These results are shown pictorially in Fig. 13. The nearly identical answers in all three cases was due to the stringent constraint on *mean_thsio2gate*. All three metrics improve with smaller values for the standard deviation of the gate

Inputs

vary	name	default	units	min	max
0	time_gox	1.600	sec*1e3	1.000	2.200
0	temp_gox	1.250	K*1e3	1.150	1.350
1	md_time_gox	0	sec*1e3	-0.1	0.1
0	sd_time_gox	0	sec*1e2	0	0.1
1	md_temp_gox	0	K*1e3	-0.1	0.1
0	sd_temp_gox	0	K*1e2	0	0.1

Coefficients

tune	name	value	min	max
1	lim_thsio2gate	4.2	3.7	4.7

Outputs

select	name	target	units	min	max	weight	rsm
Constraint	mean_thsio2gate	4.2	1e-8m	4.1	4.3		mean_thsio2gate.rsm
Target	sig_thsio2gate	1e-2	1e-8m	1.0e-2	1		sig_thsio2gate.rsm
	s2n_thsio2gate	100		1	100	1.0	s2n_thsio2gate.rsm
	ym_thsio2gate	100	%	50	100	1.0	ym_thsio2gate.rsm
Constraint	ysd_thsio2gate	0	%	0	5		ysd_thsio2gate.rsm

Opt Method Comparisons

case	md_ time_gox	md_ temp_gox	mean_ thsio2gate	sig_ thsio2gate	s2n_ thsio2gate	ym_ thsio2gate	ysd_ thsio2gate
1	-0.1	0.01087	4.300021075	0.3063008676	14.14484137	84.34461003	3.187593261
2	-0.1	0.010869	4.300007041	0.3063571793	14.13267319	84.38151297	3.187613642
3	-0.0165	-0.0041596	4.182424053	0.3191752864	13.25157406	87.1727676	3.477796522

Fig. 12. DOE/Opt setup for statistical process optimization of gate oxide thickness using FABRICS, and verified simulation results for three optimization strategies.

oxide (*sig_thsio2gate*). Yield is maximized if in addition to a decreasing variability the design is "centered"—the mean value is chosen so that most of the distribution is within the constraints of acceptability. Similarly the signal-to-noise is maximized as the mean value increases without adversely affecting the standard deviation, or the standard deviation decreases without a significant drop in the mean value. For the case where the mean value is tightly constrained (towards the center of the window in this case) all three metrics are optimized when *sig_thsio2gate* is minimized, subject to the constraints on *mean_thsio2gate*. The constraint on the mean ensured that both *s2n_thsio2gate* and *ym_thsio2gate* are strongly dominated by *sig_thsio2gate*. This is the case even when the *sig_thsio2gate* and *mean_thsio2gate* are not independent. The difference in the results between the yield maximization and the remaining metrics can be attributed to the fact that the distribution of the gate oxide thickness results in larger yield loss in the tail regions when signal to noise or standard deviation alone are optimized.

The dependence of *s2n_thsio2gate* and *ym_thsio2gate* on *mean_thsio2gate* was explored by repeating the same optimizations without the constraints on the *mean_thsio2gate*. Results (shown schematically as dashed curves in Fig. 13) indicated that both *s2n_thsio2gate* and *ym_thsio2gate* were strong functions of both *sig_thsio2gate* and *mean_thsio2gate*. Moreover, the sensitivities of the yield and signal to noise ratios to the mean and the standard deviation were significantly different. Maximizing *ym_thsio2gate* produced the same results with and without the constraint on *mean_thsio2gate*. On the other hand, the *sig_thsio2gate* minimization and *s2n_thsio2gate* maximizations produced identical results which were significantly different from the corresponding results with the constraints on *mean_thsio2gate*. The optimizations were dominated by decreasing the variability of the gate oxide thickness and the resulting process conditions produced extremely low yields, i.e., the optimizer was able to push the process conditions near the edge of the acceptability region.

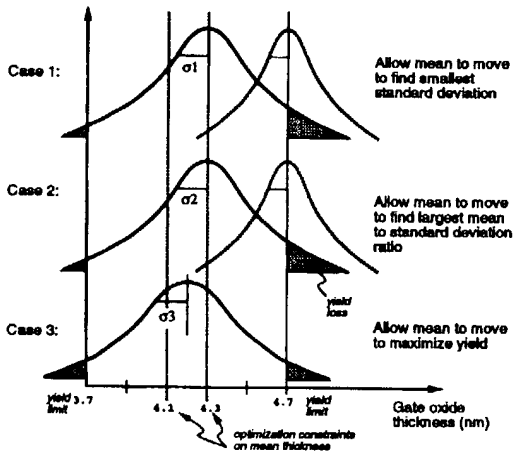

Fig. 13. Manufacturability optimization for 1) minimum standard deviation (constrained mean in solid lines, unconstrained in dashed line), 2) maximum signal to noise, and 3) maximum yield.

This example demonstrates that DOE/Opt can be used to assist in statistical process design, an integral part of design for manufacturability. Several different objective functions can be defined and the results of using different statistical criteria for optimization can be explored. We have been able to derive and calculate multi-dimensional yield by encapsulating a statistical process simulator, FABRICS. Finally, we have been able to determine the process conditions that maximize the value of the yield using the optimizer NPSOL in DOE/Opt.

VII. Conclusion

A system has been implemented to provide design of experiments, regression modeling, and optimization capability for use in Technology CAD. The system emphasizes the uniform treatment of model evaluation across numerical simulation and response surface modeling. In this fashion, effective optimization layered on top of existing simulation capability is possible. We have demonstrated the application of the system to process parameter determination, simulator tuning, process control modeling, and statistical process optimization. Future work is needed to extend the system to more fully support emerging device design and process synthesis methodologies [43].

Acknowledgment

We gratefully acknowledge the support and guidance of J. H. Chern and P. Yang in this work. The DOE/Opt system has continued to evolve with the contributions of D. Durbeck and B. Pu. Many thanks to the initial cadre of users. Particular thanks are due to A. Chatterjee for the pass transistor design example, and S. Aur for the simulator tuning example.

References

[1] D. C. Cole, E. M. Buturla, S. S. Furkay, K. Varahramyan, J. Slinkman, J. A. Mandelman, D. P. Foty, O. Bula, A. W. Strong, J. W. Park, T. D. Linton Jr., J. B. Johnson, M. V. Fischetti, S. E. Laux, P. E. Cottrell, H. G. Lustig, F. Pileggi, and D. Katcoff, "The use of simulation in semiconductor technology development," *Solid-State Electron.*, vol. 33, no. 6, pp. 591–623, 1990.

[2] D. Boning, G. Chin, R. Cottle, W. Dietrich, S. Duvall, M. Giles, R. Harris, M. Karasick, N. Khalil, M. Law, M. J. McLennan, P. K. Mozumder, L. Nackman, S. Nassif, V. T. Rajan, D. Schroeder, R. Tremain, D. M. H. Walker, R. Wang, and A. Wong, "Developing and integrating TCAD applications with the Semiconductor Wafer Representation," in *Tech. Dig. Workshop on Numerical Modeling of Processes and Devices for Integrated Circuits: NUPAD-IV*, Seattle, WA, June 1992.

[3] SPR Working Group of the CFI/TCAD Framework Group, "Semiconductor Process Representation Requirements," CFI Document TCAD-91-G-4, Apr. 1992.

[4] F. Fasching, C. Fischer, S. Selberherr, H. Stippel, W. Tuppa, and H. Read, "A PIF implementation for TCAD purposes," in *Proc. 4th Int. Conf. on Simulation of Semiconductor Devices and Processes*, Zurich, Switzerland, Sept. 1991, pp. 477–482.

[5] M. R. Simpson, "PRIDE: An integrated design environment for semiconductor device simulation," *IEEE Trans. Computer-Aided Design*, vol. CAD-10, no. 9, pp. 1163–1174, Sept. 1991.

[6] K. V. V. Gopalarao, U. DasGupta, R. Jain, D. Boning, P. K. Mozumder, and V. Chandramouli, "An integrated technology CAD system for process and device designers," in *Proc. Sixth Int. Conf. VLSI Design*, Bombay, India, Jan. 1993.

[7] J. Mar, K. Bhargavan, S. Duvall, R. Firestone, D. Lucey, S. Nandgaonkar, S. Wu, K.-S. Yu, and F. Zarbakhsh, "EASE—An application-based CAD system for process design," *IEEE Trans. Comput.-Aided Design*, vol. CAD-6, no. 6, pp. 1032–1038, Nov. 1987.

[8] K. M. Cham, S.-Y. Oh, and J. L. Moll, "Computer-aided design in VLSI device development," *IEEE J. Solid State Circuits*, vol. SC-20, no. 2, pp. 495–500, Apr. 1985.

[9] A. R. Alvarez, B. L. Abdi, D. L. Young, H. D. Weed, J. Teplik, and E. R. Herald, "Application of statistical design and response surface methods to computer-aided VLSI device design," *IEEE Trans. Comput.-Aided Design*, vol. 7, no. 2, pp. 272–288, Feb. 1988.

[10] K. K. Low and S. W. Director, "An efficient macromodeling approach for statistical IC process design," in *IEEE Int. Conf. CAD, ICCAD-88*, Santa Clara, CA, Nov. 1988, pp. 16–19.

[11] P. Cox, P. Yang, S. S. Mahant-Shetti, and P. Chatterjee, "Statistical modeling for efficient parametric yield estimation of MOS VLSI circuits," *IEEE Trans. Electron Devices*, vol. ED-32, no. 2, pp. 471–478, Feb. 1985.

[12] T. J. Sanders, K. Rekab, F. M. Rotella and D. P. Means, "Integrated circuit design for manufacturing through statistical simulation of process steps," *IEEE Trans. Semiconductor Manufacturing*, vol. 5, no. 4, pp. 368–372, Nov. 1992.

[13] D. S. Boning, Editor, "TCAD framework architecture," SRC Publication S90013, Semiconductor Research Corporation, May 1990. (Also available as CAD Framework Initiative document TCAD-90-T-1).

[14] J. K. Ousterhout, "Tcl: An embeddable command language," in *1990 Winter USENIX Conf. Proc.*, 1990.

[15] J. K. Ousterhout, "An X11 Toolkit based on the Tcl language," in *1991 Winter USENIX Conf. Proc.*, 1991.

[16] P. E. Gill, W. Murray, M. A. Saunders, and M. H. Wright, "User's guide for NPSOL (Version 4.0): A fortran package for nonlinear programming," Tech. Rep. SOL-86, Systems Optimization Lab., Stanford Univ., Jan. 1986.

[17] N. Khalil, "Framework component: Tool control language," in SRC Publication S90013, Semiconductor Research Corporation, May 1990. (Also available as CAD Framework Initiative document TCAD-90-T-1).

[18] C. P. Ho, J. D. Plummer, S. E. Hansen and R. W. Dutton, "VLSI process modeling—SUPREM-III," *IEEE Trans. Electron Devices*, vol. ED-30, no. 11, pp. 1438–1452, Nov. 1983.

[19] G. E. P. Box, W. G. Hunter, and J. S. Hunter, *Statistics for Experimenters* (Wiley Series in Probability and Mathematical Statistics). New York: John Wiley, 1978.

[20] G. E. P. Box and K. B. Wilson, "On the experimental attainment of optimum conditions," *J. Roy. Statist. Soc. B*, vol. 13, pp. 1–45, 1951.

[21] B. Wheeler, *ECHIP™ Course Text.* ECHIP Inc., 724 Yorklyn Rd., Hockessin, DE 19707, 1989.

[22] M. D. McKay, W. J. Conover, and R. J. Beckman, "A comparison of three methods for selecting values of input variables in the analysis of output from a computer code," *Technometrics*, pp. 239–245, 1979.

[23] A. B. Owen, "Controlling correlations in latin hypercube samples," Tech. Rep., Dept. of Statist. and Center for Integrated Syst., Stanford Univ., 1991.

[24] R. L. Iman and M. J. Shortencarier, "A FORTRAN 77 program and user's guide for the generation of latin hypercube and random samples for use with computer models," technical report NUREG/CR-3624 SAND83-2365 RG, Sandia National Labs., Mar. 1984.

[25] D. F. Swayne, D. Cook, and A. Buja, "Users Manual for XGobi, a Dynamic Graphics Program for Data Analysis Implemented in the X Window System," Tech. Memo., Bellcore, 2.0 ed., 1991.

[26] G. E. P. Box and N. R. Draper, *Empirical Model Building and Response Surfaces* (Wiley series In Probability and Mathematical Statistics). New York: John Wiley, 1987.

[27] N. R. Draper and H. Smith, *Applied Regression Analysis* (Wiley Series in Probability and Mathematical Statistics). New York: John Wiley, 1981.

[28] R. A. Becker, J. M. Chambers, and A. R. Wilks, *The NEW S Language* (Computer Science Series). Pacific Grove, CA: Wadsworth & Brooks/Cole, 1988.

[29] Statistical Sciences, Inc., Seattle, WA, *S-Plus Reference Manual*, 3.0 ed., Sept. 1991.

[30] Technology Modeling Associates, Inc., 3950 Fabian Way, Palo Alto, CA 94303-4605, *TMA MEDICI*, 1992.

[31] Technology Modeling Associates Inc., Third Floor, 300 Hamilton Avenue, Palo Alto, CA 94301-9951, *TMA PISCES-2B Version 9033*, 1991.

[32] J. A. Seitchick, C. F. Machala, III, and P. Yang, "The determination of SPICE Gummel–Poon parameters by a merged optimization-extraction technique," in *BCTM Proc.*, 1989.

[33] D. J. Collins, A. J. Strojwas and D. White, "A CFD model for PECVD of silicon nitride," in *Proc. 3rd Int. Symp. Process Physics and Modeling*, Honolulu, HI, Electrochemical Society, May 1993.

[34] P. K. Mozumder, S. Saxena, and D. J. Collins, "A monitor wafer based controller for PECVD silicon nitride process on AMT 5000,"

in *Advanced Semiconductor Manufacturing Conf. and Workshop*, 1993.
[35] G. E. P. Box, "Studies in quality improvement: Signal to noise ratios, performance criteria and statistical analysis," Tech. Rep. 11, Center for Quality and Productivity Improvement, Univ. of Wisconsin-Madison, 1986.
[36] G. Taguchi, *Introduction to Quality Engineering: Designing Quality into Product and Processes* (UNIPUB). White Plains, NY: Kraus Int. Pub., 1986.
[37] G. Taguchi and M. S. Phadke, "Quality engineering through design optimization," *IEEE Commun. Society Global Telecommun. Conf.*, Nov. 1984.
[38] P. K. Mozumder, "Statistical quality control of VLSI fabrication processes," Ph.D. thesis, Carnegie-Mellon Univ., Mar. 1986.
[39] S. Sharifzadeh, J. R. Koehler, A. B. Owen, and J. D. Shott, "Using simulators to model transmitted variability in IC manufacturing," *IEEE Trans. Semiconductor Manufacturing*, vol. 2, no. 3, pp. 82–93, Aug. 1989.
[40] S. R. Nassif, A. J. Strojwas, and S. W. Director, "FABRICS II: A statistically based IC fabrication process simulator," *IEEE Trans Comput.-Aided Design*, vol. CAD-3, no. 1, pp. 40–46, Jan. 1984.
[41] A. J. Strojwas and S. W. Director, "The process engineer's workbench," *IEEE J. Solid State Circuits*, vol. 23, no. 2, pp. 377–386, 1988.
[42] C. J. Spanos, "Statistical parameter extraction for IC process characterization," Ph.D. thesis, Carnegie-Mellon Univ., 1985. Res. Rep. No. CMUCAD-85-46.
[43] M. Rodder, A. Chatterjee, D. Boning, and I. C. Chen, "Transistor design with TCAD tuning and device optimization for process/device synthesis," in *Proc. Tech. Papers, 1993 Int. Symp. on VLSI Technol., Syst., and Applicat.*, Taipei, Taiwan, 1993. Paper A7, pp. 29–33.

Duane S. Boning (S'83–M'91) received the S.B. degrees in electrical engineering and in computer science in 1984, and the S.M. and Ph.D. degrees in electrical engineering in 1986 and 1991, respectively, all from the Massachusetts Institute of Technology.

He was an NSF Fellow from 1984 to 1989, and an Intel Graduate Fellow in 1990. From 1991 to 1993, he was a Member Technical Staff at the Texas Instruments Semiconductor Process and Design Center in Dallas, TX, where he worked on semiconductor process representation, process/device simulation tool integration, and statistical modeling and optimization. He is currently an Assistant Professor in the Electrical Engineering and Computer Science Department at MIT. His research interests include process control, process synthesis, tools and frameworks for semiconductor process and device design, and the computer integrated manufacturing of integrated circuits.

Dr. Boning is an Associate Editor for the IEEE TRANSACTIONS ON SEMICONDUCTOR MANUFACTURING, and has served as chairman of the CFI/Technology CAD Framework Semiconductor Process Representation Working Group. He is a member of Eta Kappa Nu, Tau Beta Pi, Sigma Xi, and the Association of Computing Machinery.

Purnendu K. Mozumder (S'88–M'89) received the B. Tech Degree in electrical engineering from the Indian Institute of Technology in 1984, and the M.S. and Ph.D. degrees in electrical and computer engineering in 1986 and 1989, respectively, from Carnegie Mellon University.

He is currently a member of the technical staff at the Semiconductor Process and Design Center of Texas Instruments Inc., Dallas, TX. He is presently working on design for manufacturing, TCAD, and statistical process control for VLSIC fabrication. He has been actively involved in the Microelectronics Manufacturing Science and Technology (MMST) project at TI. His interests include stochastic and adaptive control, statistical process and device simulation, TCAD Framework development, and application of statistical and numerical techniques to IC design and manufacturing methodologies.

Dr. Mozumder is presently serving as a Guest Editor of the IEEE TRANSACTIONS ON SEMICONDUCTOR MANUFACTURING special issue on MMST.